

3100000011440



MILIK PERPUSTAKAAN  
INSTITUT TEKNOLOGI  
SEPULUH - NOPEMBER

391

## TUGAS AKHIR (NA.1701)

### KOMPUTERISASI STANDAR PENYIMPANGAN DAN PERBAIKAN PADA PROSES PRODUKSI KAPAL

RSPe  
623.810285  
Feb  
k-1  
1998



OLEH :

29-6-99  
H  
8649

TEDDY KRISHNA FEBION  
4192 100 019

**JURUSAN TEKNIK PERKAPALAN  
FAKULTAS TEKNOLOGI KELAUTAN  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
1998**



# JURUSAN TEKNIK PERKAPALAN

## FAKULTAS TEKNOLOGI KELAUTAN ITS

### SURAT KEPUTUSAN TUGAS AKHIR (NA 1701)

No. : 39 /PT12.FTK2/M/199 7

Nama Mahasiswa : Teddy Krisna Febion .....  
Nomor Pokok : 4192100019 .....  
Tanggal diberikan tugas : 16 Maret 1997 .....  
Tanggal selesai tugas : 26 Juli 1997 .....  
Dosen Pembimbing : 1. Ir. S. o. e. j. i. t. n. e .....  
2. Ir. Sjarief Widjaja, Ph.D. ....

Uraian / judul tugas akhir yang diberikan :

KOMPETERISASI STANDAR PENYIMPANGAN DAN PERBAIKAN PADA PROSES PRODUKSI KAPAL

sOn

Surabaya, 31 Maret 1997

Jurusan Teknik Perkapalan FTK-ITS



Tembusan :

1. Yth. Dekan FTK-ITS.
2. Yth. Dosen Pembimbing.
3. Arsip.

Ir. Koesowo Sastro Wiyono  
NIP. 130 687 430.

## LEMBAR PENGESAHAN

Tugas Akhir

NA 1701

### KOMPUTERISASI STANDAR PENYIMPANGAN DAN PERBAIKAN PADA PROSES PRODUKSI KAPAL

Disusun oleh :

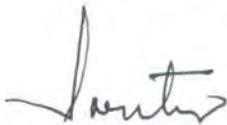
Teddy Krishna Febion

NRP 4192 100 019

Telah diperiksa dan dinyatakan siap untuk diujikan  
pada tanggal      Maret 1998

Surabaya, 23 February 1998

Dosen Pembimbing I



Ir. Soejitno

Dosen Pembimbing II



DR. Ir. R. Sjarief Widjaja



## ABSTRAK

*Pada proses pembangunan kapal di beberapa galangan nasional masih membutuhkan waktu yang lama serta mutu konstruksi yang masih rendah. Hal ini disebabkan oleh besarnya volume pekerjaan ulang (rework) akibat penyimpangan bentuk dan dimensi komponen badan kapal di luar toleransi yang diperkenankan.*

*Penyimpangan yang sering terjadi pada proses pembangunan kapal ini sangat dipengaruhi oleh deformasi material, peralatan bending/ potong/ las yang digunakan, kualifikasi tenaga kerja dan urutan pengerjaannya. Disamping itu, permasalahan alat ukur yang tepat, metode pengukuran dan teknik perbaikan atas penyimpangan bentuk dan dimensi komponen tersebut masih membutuhkan penjabaran yang lebih lanjut.*

*Berdasarkan permasalahan tersebut, maka perlu dibuat data base mengenai segala bentuk penyimpangan dan perbaikan, dimana data ini sangat membantu dalam menentukan penyimpangan yang terjadi yang kemudian penyimpangan tersebut dapat diidentifikasi menurut jenis penyimpangannya, baik penyimpangan material, peralatan, penyimpangan dimensi dan sebagainya.*

*Sebagai hasil dari database ini maka akan dapat memudahkan dan mempercepat dalam menentukan perbaikan apa yang perlu dilakukan. Sehingga akan mempersingkat waktu dalam proses produksi.*



## KATA PENGANTAR

Dengan mengucapkan syukur atas segala rahmat serta petunjuk Tuhan Yang Maha Kuasa, maka kami dapat menyelesaikan tugas akhir ( NA 1701 ) ini dengan judul " Komputerisasi Standar Penyimpangan Dan Perbaikan Pada Proses Produksi Kapal". Tugas akhir ini kami susun untuk melengkapi persyaratan kurikuler guna menyelesaikan studi di jurusan Teknik Perkapalan, Fakultas Teknologi Kelautan, Institut Teknologi Sepuluh Nopember Surabaya.

Walaupun kami telah berusaha untuk menyelesaikan penulisan tugas akhir ini sesempurna mungkin, namun sebagai manusia biasa kami tetap memiliki kelemahan dan keterbatasan kemampuan sehingga mungkin masih banyak ditemukan yang tidak kami sengaja dalam penulisan Tugas Akhir ini. Karena itu kami sangat mengharapkan adanya masukan-masukan baik berupa saran maupun koreksi yang berguna untuk lebih menyempurnakan tugas akhir ini sehingga pada gilirannya dapat lebih memberikan manfaat bagi pihak-pihak yang memerlukannya.

Selama melakukan penelitian dan penulisan tugas akhir ini, penulis telah menerima bimbingan, bantuan, nasehat, dan dorongan yang sangat besar dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada yang terhormat :

1. Bapak Ir. Koestowo Sastro Wiyono, selaku Ketua Jurusan Teknik Perkapalan, Fakultas Teknologi Kelautan, Institut Teknologi Sepuluh Nopember Surabaya.
2. Bapak Ir. Soejitno, selaku Dosen Pembimbing yang selalu memberi

petunjuk dan saran selama penulis menyelesaikan Tugas Akhir.

3. Bapak Ir. Sjarief Widjaja, PhD. selaku Dosen Pembimbing Tugas Akhir yang tanpa kenal lelah dan waktu telah memberikan bimbingan dan petunjuk yang sangat bermanfaat bagi penulis.
4. Bapak Ir. Imam, Kepala Divisi QA/QC PT. PAL Indonesia
5. Bapak Fatchul, Kepala QA/QC yang turut membantu dalam pelaksanaan pengambilan data di PT. PAL Indonesia.
6. Kedua Orang tua, Kakak dan adik tercinta atas segala doa bagi penulis untuk penyelesaian Studi dan Tugas Akhir ini.
7. Yang tersayang Lissa Widya Kussuma yang selalu memberikan doa, semangat dan dorongan selama penulis menyelesaikan Studi dan Tugas Akhir.
8. Bapak-bapak dosen yang telah memberikan bekal ilmu pengetahuan selama penulis menempuh masa perkuliahan.
9. Antok, Andi, Dudut, Riza, yang tergabung dalam *Grup Badai* yang selalu memberi semangat dan dorongan hingga terselesaikannya Tugas Akhir ini
10. Anton dan keluarga, Opieq, Anis, Taufik atas segala bantuannya.
11. Rekan – rekan dan semua pihak yang telah banyak membantu hingga terselesaikannya Tugas Akhir ini.

Surabaya, 25 Februari 1998

Penulis



## DAFTAR ISI

JUDUL.....	I
COPY SURAT KEPUTUSAN TUGAS AKHIR.....	II
LEMBAR PENGESAHAN .....	III
ABSTRAK.....	IV
KATA PENGANTAR.....	V
DAFTAR ISI.....	VII
DAFTAR GAMBAR.....	IX
BAB I.....	1
PENDAHULUAN.....	1
I.1. LATAR BELAKANG.....	1
I.2. TUJUAN PENULISAN .....	2
I.3. MANFAAT PENULISAN .....	2
I.4. BATASAN MASALAH.....	3
I.5. METODOLOGI PENULISAN.....	4
I.6 KESIMPULAN AWAL.....	5
BAB II.....	6
TEKNOLOGI PEMBANGUNAN KAPAL .....	6
II.1. PERKEMBANGAN TEKNOLOGI PEMBANGUNAN KAPAL .....	6
II.2. FAKTOR-FAKTOR YANG MEMPENGARUHI PROSES PRODUKSI.....	10
Material.....	10
Tenaga Kerja.....	11
Metode Produksi.....	12
Peralatan Produksi.....	13
II.3. TAHAPAN PROSES PRODUKSI.....	13
II.3.1. TAHAP PERSIAPAN PRODUKSI.....	14
II.3.2. TAHAP FABRIKASI .....	14
II.3.3. TAHAP SUB ASSEMBLY.....	19
II.3.4. TAHAP ASSEMBLY.....	20
II.3.5. TAHAP ERECTION.....	21
II.4. PENGAWASAN PRODUKSI KAPAL.....	23
a. Tahap Persiapan, .....	25
b. Tahap Fabrikasi .....	25
c. Tahap Perakitan ( sub-assembly dan assembly ) .....	25
d. Tahap Penyambungan ( Erection ) .....	25
BAB III .....	30
PENYIMPANGAN DIMENSI DAN BENTUK PADA PROSES PRODUKSI KAPAL.....	30
III.1. TEORI DASAR STANDAR PENYIMPANGAN DIMENSI DAN BENTUK .....	30
a. Standarisasi Excess.....	30
b. Standarisasi untuk Allowance penyusutan.....	30
c. Standarisasi garis dasar dan titik pertemuan .....	31



d. Standarisasi produksi.....	31
e. Standarisasi prosedur pemeriksaan.....	31
f. Standarisasi informasi A/C.....	31
III.2. JENIS PENYIMPANGAN DIMENSI DAN BENTUK PADA PROSES PRODUKSI KAPAL ....	46
1. PROSES FABRIKASI.....	47
2. PROSES SUB – ASSEMBLY DAN ASSEMBLY.....	47
3. PROSES ERECTION.....	48
III.3 TEKNIK PERBAIKAN PADA PENYIMPANGAN DIMENSI DAN BENTUK.....	54
Misalignment ( Ketidak lurusan sambungan Butt Join ).....	54
Misalignmen ( ketidak lurusan pada sambungan Fillet ).....	55
Gap Atau Celah.....	55
Gap atau celah untuk sambungan fillet.....	57
Leg Length.....	58
Tinggi, Lebar, Dan Sudut Pengelasan.....	59
Under Cut.....	59
<b>BAB IV .....</b>	<b>60</b>
<b>STUDI KASUS PENYIMPANGAN DIMENSI DAN BENTUK PADA PROSES PRODUKSI KAPAL .....</b>	<b>60</b>
IV.1 PERSIAPAN SURVEI DAN PENGAMBILAN DATA.....	60
IV.2 HASIL SURVEI DAN INTERVIEW TENTANG PENYIMPANGAN DIMENSI DAN BENTUK.....	61
<b>BAB V.....</b>	<b>63</b>
<b>PEMBUATAN MODEL PENYIMPANGAN DIMENSI DAN BENTUK.....</b>	<b>63</b>
V.1. HASIL ANALISA DARI STANDAR JSQS DAN SURVAI LAPANGAN .....	63
1. Tahap Fabrikasi .....	64
2. Tahap Sub- Assembly dan Assembly.....	66
3. Tahap Erection.....	70
V.2. MODEL ALAT BANTU UNTUK STANDARISASI PENYIMPANGAN DAN INSTRUKSI PERBAIKAN .....	77
V.3. FLOWCHART PROGRAM KOMPUTER UNTUK IDENTIFIKASI PENYIMPANGAN DAN INSTRUKSI PERBAIKAN.....	80
<b>BAB VI.....</b>	<b>82</b>
<b>PEMBUATAN PROTOTYPE PROGRAM PENYIMPANGAN BENTUK DAN INSTRUKSI PERBAIKAN .....</b>	<b>82</b>
VI.1. ALASAN PEMILIHAN BAHASA PEMROGRAMAN.....	82
VI.2. TAHAPAN KONSULTASI.....	87
VI.3. HASIL KONSULTASI .....	94
<b>BAB VII.....</b>	<b>95</b>
<b>DISKUSI DAN REKOMENDASI.....</b>	<b>95</b>
<b>BAB VIII .....</b>	<b>97</b>
<b>KESIMPULAN DAN SARAN.....</b>	<b>97</b>
<b>DAFTAR PUSTAKA .....</b>	<b>99</b>
<b>LAMPIRAN</b>	

## DAFTAR GAMBAR

GAMBAR 2.1	TAHAPAN PERKEMBANGAN TEKNOLOGI .....	6
GAMBAR 2.2.	KOMPONEN PRODUCT-ORIENTED WORK BREAKDOWN STRUCTURE .....	8
GAMBAR 2.3	PROSES PRODUKSI BENGKEL FABRIKASI .....	16
GAMBAR 3.1	GRAFIK HUBUNGAN PENGARUH TEBAL TERHADAP PENYUSUTAN MELINTANG .....	35
GAMBAR 3.2.	GRAFIK HUBUNGAN ANTARA PENYUSUTAN DENGAN BERAT LOGAM LAS .....	35
GAMBAR 3.3	GRAFIK PENGARUH TEBAL PELAT DAN CELAH AKAR TERHADAP PENYUSUTAN .....	36
GAMBAR 3.4	GRAFIK HUBUNGAN ANTARA PERUBAHAN SUDUT DENGAN TEBAL PELAT .....	38
GAMBAR 3.5	GRAFIK KONDISI PENGELASAN .....	39
GAMBAR 3.6	GRAFIK HUBUNGAN ANTARA JUMLAH PENGELASAN TERHADAP PENYIMPANGAN PADA SAMBUNGAN T .....	40
GAMBAR 3.7	GRAFIK HUBUNGAN ANTARA BERAT LOGAM LAS DENGAN PERUBAHAN SUDUT .....	41
GAMBAR 3.8	GRAFIK HUBUNGAN ANTARA PROSEDUR PENGELASAN DAN PERUBAHAN WUJUD .....	42
GAMBAR 3.9	PEMBEBANAN PADA PROSES PENGELASAN .....	45
GAMBAR 5.1	MODEL STANDARISASI PENYIMPANGAN DAN PERBAIKAN .....	78
GAMBAR 5.2	PERBEDAAN PROSES MANUAL DAN KOMPUTERISASI STANDAR PENYIMPANGAN .....	79
GAMBAR 5.3	ALUR PROGRAM DATABASE .....	80
GAMBAR 5.4	ALUR PROGRAM IDENTIFIKASI PENYIMPANGAN .....	81
GAMBAR 6.1	TAMPILAN UTAMA PROTOTYPE PROGRAM .....	87
GAMBAR 6.2	PEMERIKSAAN TAIAP FABRIKASI .....	88
GAMBAR 6.3	PENYIMPANGAN PADA FLIENS MEMANJANG .....	88
GAMBAR 6.4	LOKASI DAN STANDAR YANG DIJINKAN .....	89
GAMBAR 6.5	PEMERIKSAAN TAHAP SUB ASSEMBLY .....	89
GAMBAR 6.6	LOKASI DAN STANDAR YANG DIJINKAN .....	90
GAMBAR 6.7	LOKASI DAN STANDAR PENYIMPANGAN YANG DIJINKAN .....	90
GAMBAR 6.8	JENIS DAN LOKASI PENYIMPANGAN .....	91
GAMBAR 6.9	LOKASI PENYIMPANGAN .....	91
GAMBAR 6.10.	STANDAR YANG DIJINKAN .....	92
GAMBAR 6.11	LOKASI DAN JENIS PENYIMPANGAN .....	92
GAMBAR 6.12	LOKASI PENYIMPANGAN .....	93
GAMBAR 6.13	STANDAR PENYIMPANGAN YANG DIJINKAN .....	93
GAMBAR 6.14	FORM INSTRUKSI PERBAIKAN .....	94



# BAB I

## PENDAHULUAN



### I.1. LATAR BELAKANG

Pada saat ini perkembangan dunia perkapalan semakin maju dan cepat berkembang sehingga persaingan antar galangan semakin ketat. Masing-masing galangan kapal berusaha untuk meningkatkan produktifitasnya, serta menekan biaya produksi seminim mungkin.

Proses pembangunan kapal di galangan Nasional, persyaratan mutu konstruksi masih sulit dipenuhi dan dianggap mahal. Karena pencapaian mutu harus dilakukan melalui proses kerja ulang (*rework*). Sehingga terjadi penambahan jam orang dan material yang cukup besar. Hal ini dirasakan sebagai suatu jumlah yang cukup berarti bagi usaha peningkatan daya saing galangan kapal.

Pekerjaan dalam pembangunan kapal, merupakan pekerjaan yang sulit. Ini dikarenakan bagian yang dibangun merupakan komponen yang kompleks dan mempunyai ukuran yang relatif besar dan mempunyai bobot yang berat. Sehingga sering ditemui kesalahan dikarenakan kesalahan dalam pemasangan elemen-elemen penyusun konstruksi ataupun kemungkinan adanya deformasi.

Oleh karena itu setiap proses produksi pada perusahaan galangan kapal diperlukan suatu teknik dalam berproduksi yang benar, ini dimaksudkan agar galangan tersebut dapat melakukan proses produksi yang benar secara efektif



dan berkualitas serta dapat tepat waktu, sehingga galangan tersebut dapat bersaing dengan perusahaan-perusahaan lainnya.

Untuk mencapai maksud dan tujuan tersebut maka diperlukan suatu analisa teknis guna menekan sekecil mungkin kesalahan-kesalahan yang terjadi di setiap proses pengerjaan baik mulai dari proses perencanaan sampai pada proses *assembly*.

Dengan pembuatan paket komputer diharapkan akan sangat membantu di dalam pelaksanaan proses produksi dalam mengidentifikasi suatu jenis penyimpangan untuk selanjutnya dihasilkan proses perbaikan yang dapat dilakukan untuk memperbaiki penyimpangan tersebut.

## I.2. TUJUAN PENULISAN

Adapun tujuan penulisan dalam pembuatan tugas akhir ini adalah :

1. Pembuatan suatu *Prototype Program Database* Instruksi Perbaikan Penyimpangan Proses Produksi Kapal.
2. *Standarisasi* penyimpangan dan perbaikan pada produksi kapal
3. *Identifikasi* bentuk penyimpangan pada produksi kapal.
4. Adanya kemungkinan penerapan program *database* instruksi perbaikan penyimpangan proses produksi kapal untuk meningkatkan kecepatan dan efisiensi pekerjaan perbaikan.

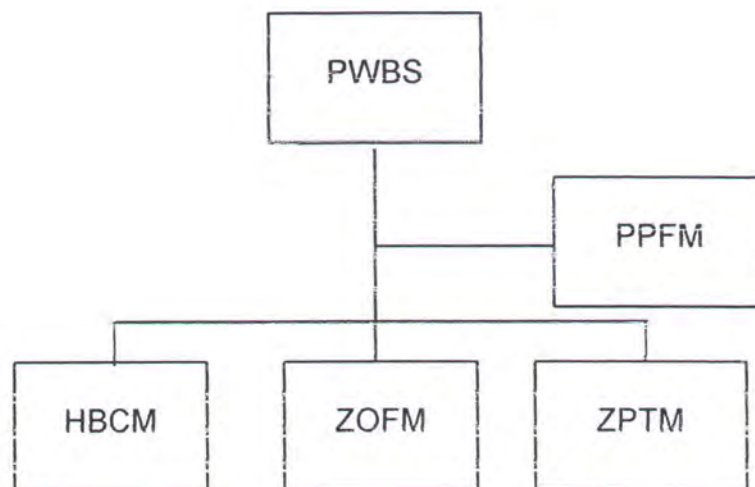
## I.3. MANFAAT PENULISAN

Dalam pelaksanaan proses produksi pembangunan kapal, kondisi di lapangan masih sering terjadi kesalahan karena berbagai faktor, sehingga sering harus dilakukan pekerjaan ulang ( *rework* ).

banyak dilakukan di bengkel dengan kondisi lingkungan kerja yang lebih nyaman. Pekerjaan pengelasan sudah mulai dilakukan dengan *mesin las semi/otomatis* dengan posisi *down-hand*. Blok-blok dapat dibalik atau diputar untuk menghindari dari pengelasan dengan posisi *overhead*. Pada tahapan ini, *kontrol dimensi* dan *bentuk blok* sudah menjadi penting terutama pada daerah sambungan blok.

- *Process Lane Construction and Zone Outfitting*

Perkembangan teknologi pembangunan kapal modern telah dimulai pada tahap ini, dimana konsep *Group Technology* sudah diterapkan dalam proses produksi badan kapal dan pekerjaan O/F. Dengan konsep ini proses pembangunan kapal sudah berorientasi pada produk atau dikenal dengan *Product-Oriented Work Breakdown Structure (PWBS)*, dengan pengelompokan lingkup pekerjaan seperti terlihat pada *Gambar 2.2*.



*Gambar 2.2. Komponen Product-Oriented Work Breakdown Structure*

HBCM	: Hull Block Construction Method
ZOFC	: Zone Outfitting Method
ZPTM	: Zone Painting Method
PPFM	: Pipe Piece Family Manufacturing



Metode ini secara sistematis mengklasifikasikan *produk antara* (*interim-products*) menurut kelompok yang memiliki kesamaan dalam proses produksinya, misalnya *process lane* untuk : (i) perakitan bentuk datar, (ii) perakitan bentuk lengkung beraturan, dan (iii) perakitan bentuk lengkung tak beraturan dan kompleks. Hal ini sudah menuntut keteraturan dalam penataan sumberdaya produksi, misalnya : peralatan/mesin, tenaga kerja, dan material. Demikian halnya dengan pekerjaan O/F, dimana sudah dilakukan secara paralel berdasarkan *region/zone*, dan tidak lagi berdasarkan sistem fungsionalnya. Pekerjaan O/F, dengan konsep *Advanced Outfitting*, sudah dibagi dalam 3 tahap yaitu : *on-unit*, *on-block*, dan *on-board*.

Pada tahapan ini, proses produksi sudah terkonsentrasi pada bengkel-bengkel dan volume pekerjaan di *building berth* semakin kecil. Penggunaan teknologi ini sudah mensyaratkan diterapkannya *Sistem Accuracy Control*, di setiap proses produksi untuk menjamin *ketepatan dimensi* dan *bentuk blok* (pelat, *profile*, perpipaan, dan *inner-parts* lainnya yang ada pada daerah sambungan blok).

- *Integrated Hull Construction, Outfitting, and Planning (IHOP)*

Tahapan keempat ini adalah teknologi yang paling *mutakhir* dalam pembangunan kapal, dimana proses pembuatan badan kapal sudah *diintegrasikan* secara maksimal dengan pekerjaan *outfitting* dan *pengecatan* pada setiap *zone/area/stage*, seperti halnya yang telah dilakukan oleh IHI-Shipyards, Japan. Teknologi ini merupakan pengembangan dari teknologi produksi tahapan ketiga, melalui peningkatan potensi sumberdaya galangan



secara *menyeluruh, seimbang dan terintegrasi*, disertai dengan semakin sempurnanya standar-standar kerja dan konsistensi ketepatan proses produksi yang sangat tinggi.

Teknologi IHOP mensyaratkan suatu *Build Strategy* yang matang dan kemampuan *planning and schedulling* yang sangat tinggi dan rasionil, berdasarkan kondisi potensi sumberdaya galangan yang ada. Pada tahap ini, Sistem Accuracy Control bukan hanya mutlak dilaksanakan secara sempurna dan menyeluruh, tetapi juga menuntut kesempurnaan *design engineering* dan *standar-standar* kerjanya. Dengan demikian, penyambungan seluruh bagian konstruksi antara blok yang satu dengan blok lainnya mencapai tingkat ketepatan yang sangat tinggi atau penyimpangan-penyimpangan yang terjadi masih berada pada batas toleransi yang telah ditentukan, sehingga volume pekerjaan pada *building berth* menjadi sangat kecil.

## II.2. FAKTOR-FAKTOR YANG MEMPENGARUHI PROSES PRODUKSI

Pada proses pembangunan kapal terdapat beberapa faktor utama yang sangat mempengaruhi kualitas hasil produksi antara lain :

### Material

Pada pembangunan kapal material yang dibutuhkan sangat mempengaruhi jalannya proses produksi. Untuk material logam baja diperlukan proses pengerjaan meliputi pemotongan, pembentukan, penggabungan pelat dan profil. Berdasarkan proses pengerjaan dari masing-masing material yang berbeda maka peralatan yang dibutuhkan, kualifikasi

Manfaat dari penulisan tugas akhir ini adalah :

- Memberikan masukan dan saran teknis guna peningkatan produktifitas galangan kapal.
- Menjaga agar mutu perbaikan penyimpangan selalu tetap terjaga.
- Memperkecil terjadinya ketergantungan kepada perseorangan dalam menentukan instruksi perbaikan.
- Membantu dalam menunjang kemajuan teknologi di bidang perkapalan yang bermanfaat bagi masyarakat pemakainya.

#### I.4. BATASAN MASALAH

- Program *database* yang dibuat hanyalah merupakan *prototype* program demonstrasi.
- Pada penulisan tugas akhir ini proses produksi yang dilakukan dianggap sudah memenuhi standart kerja dan peraturan yang berlaku.
- Standarisasi penyimpangan dan perbaikan hanya dilakukan pada konstruksi Lambung.
- Bentuk penyimpangan dilihat berdasar kejadian selama dilakukan proses produksi pada komponen setiap tahap dan tidak didasarkan pada jenis kapal.
- *Output* yang diharapkan dari *prototype* program ini adalah Instruksi Perbaikan Penyimpangan.

## I.5. METODOLOGI PENULISAN

Metodologi penelitian yang akan dilaksanakan pada pembahasan masalah adalah :

1. Studi Literatur :

Berdasarkan literatur-literatur yang mempunyai relevansi dengan penulisan tugas akhir ini dan dapat sebagai teori penunjang, serta pembuatan program *database* untuk sistem ini.

2. Studi Lapangan :

Untuk mendapatkan data-data lapangan serta survei yang dimaksudkan untuk memperoleh data dan keterangan yang nyata dan valid sehingga menambah kelengkapan dan kebenaran dari penulisan tugas akhir ini.

3. Analisa Data :

Dari data yang telah didapat dari literatur dan studi lapangan, dilakukan identifikasi bentuk penyimpangan berdasarkan standar ***Japanese Shipbuilding Quality Standard ( JSQS )***.

4. Penyusunan Program Komputer :

Dalam tahap ini akan dikumpulkan seluruh konsep program, bab-bab masalah dan aspek kontrol dari program *database* yang dibuat.

Program *database* disusun berdasarkan seluruh data- data bentuk penyimpangan konsep yang telah disusun.

5. Pengujian Program :

Dalam tahap ini akan diperiksa apakah program telah mampu berjalan sesuai dengan yang diharapkan.



## 1.6 KESIMPULAN AWAL

Standarisasi penyimpangan dan perbaikan dapat dilakukan pada proses produksi kapal dikarenakan pada proses produksi kapal masih sering terjadi penyimpangan yang sudah barang tentu akan sangat mempengaruhi berlangsungnya proses produksi.

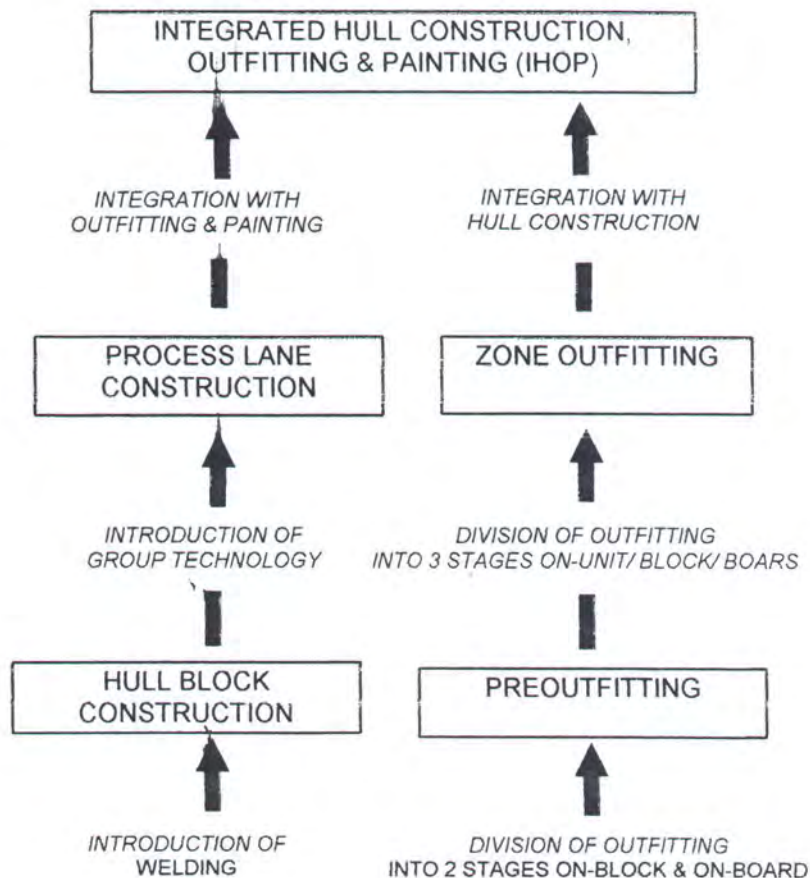
Dengan adanya standarisasi hasil dari instruksi perbaikan dapat terjaga dengan baik dan setiap orang dapat cepat memberikan instruksi perbaikan apabila dalam pelaksanaan proses produksi kapal ditemukan penyimpangan – penyimpangan produksi.

## BAB II

### TEKNOLOGI PEMBANGUNAN KAPAL

#### II.1. PERKEMBANGAN TEKNOLOGI PEMBANGUNAN KAPAL

Teknologi pembangunan kapal mulai berkembang pesat setelah Perang Dunia II, yang disponsori oleh *IHI-Shipyards*, Japan. Pengembangan tersebut didasarkan pada usaha peningkatan efisiensi pada proses produksi atau produktivitas pada pembangunan beberapa kapal Tanker berukuran besar. Menurut *Chirillo*, perkembangan teknologi produksi kapal dibagi menjadi *empat tahapan*, berdasarkan teknologi yang digunakan pada proses produksinya, seperti pada *Gambar 2.1*



Gambar 2.1 : Tahapan Perkembangan Teknologi



- *Conventional Construction and Outfitting*

Tahapan pertama ini merupakan teknologi produksi kapal, yang berorientasi pada *sistem atau fungsi* yang ada di kapal dan volume pekerjaan hampir seluruhnya dilakukan pada *building berth*. Metode ini dimulai dengan peletakan lunas, kemudian pemasangan gading, kulit. Dst sampai ke bangunan atas dan terakhir pekerjaan *outfitting (O/F)*. Pekerjaan O/F dilakukan sistem per sistem, antara lain : pemasangan ventilasi, perpipaan, permesinan, perlistrikan, dll. Metode tersebut merupakan teknologi paling *konvensional* dan tingkat produktivitasnya sangat rendah, karena semua lingkup pekerjaan memiliki ketergantungan yang tinggi satu sama lain, sehingga membutuhkan waktu yang sangat lama. Selain itu, mutu hasil pekerjaan sangat rendah karena hampir seluruh pekerjaan dilakukan secara manual dan lingkungan kerja yang tidak mendukung (tidak nyaman, posisi kerja yang sulit).

- *Hull Block Construction Method & Pre-Outfitting*

Tahapan ini mulai dikembangkan sejak dikenalkannya teknologi pengelasan pada pembangunan kapal, dimana kapal sudah dibuat dalam bentuk seksi-seksi dan blok-blok, kemudian disambung satu sama lain di *building berth*. Selain itu, beberapa pekerjaan O/F sudah mulai dilakukan pada blok atau badan kapal yang sudah jadi. Dengan metode ini, *steel throughput* meningkat dan mutu hasil pekerjaan lebih baik, karena volume pekerjaan pada *building berth* menjadi berkurang, dan pekerjaan pengelasan

tenaga kerja, tata letak bengkel dan prosedur kerja yang digunakan akan sangat berbeda. Sehingga ada beberapa hal yang harus diperhatikan di dalam pemakaian material yaitu :

- *Spesifikasi Teknik Material*

meliputi : sifat - sifat mekanik, seperti kekuatan tarik, tekan, kemuluran dan beberapa sifat kimia kandungan logamnya.

- *Ukuran / dimensi*

meliputi : ketebalan, panjang, lebar dan kelurusannya.

- *Pabrik pembuat / produsen*

meliputi : kualifikasi dari pabrik pembuat dilihat dari cacatan mutu produk yang dihasilkan.

- *Lingkungan / cuaca / temperatur / kelembaban*

Yaitu merupakan media dimana material tersebut akan dikerjakan atau diproses. Hal ini tergantung dari tempat proses pekerjaan dilakukan. Misalnya untuk di Indonesia maka kondisi lingkungannya akan sangat berbeda dengan Eropa.

### **Tenaga Kerja**

Kualifikasi tenaga kerja pada proses produksi akan sangat berbeda, tergantung dari jenis proses pengerjaan juga jenis material yang akan dikerjakan. Sedangkan untuk pemakaian tenaga kerja dapat diklasifikasikan dari segi :

- *Pendidikan formal / non formal*



Meliputi latar belakang pendidikan seperti STM, Politeknik, dan *kualifikasi* khusus seperti *Welder Qualification grade G3*, dll. Hal ini sangat berpengaruh terhadap pemilihan tenaga kerja.

- *Pengalaman dan masa kerja*

Seorang tenaga kerja yang telah mempunyai pengalaman dalam bidang yang sesuai akan sangat membantu pada proses pelaksanaan pekerjaan.

- *Ketrampilan / Skill ( kualifikasi )*

Ketrampilan khusus yang dimiliki oleh tenaga kerja akan sangat membantu didalam proses pekerjaan. Misalnya pada teknologi pengelasan logam tipis, membending pelat ,dan sebagainya.

- *Sikap / Karakter*

Sikap dan karakter setiap tenaga kerja akan sangat membantu pada pelaksanaan proses produksi sehingga akan menciptakan iklim kerja sesuai dengan yang diinginkan.

### **Metode Produksi**

Hal-hal yang harus diperhatikan pada penetapan metode produksi adalah yang berkaitan langsung dengan tugas masing-masing tenaga kerja, diantaranya :

- Standar dan Prosedur kerja ( *Operating Procedure Agreement* )
- Urutan pekerjaan ( *Operating Instruction* )
- Peralatan / perlengkapan keselamatan kerja

### Peralatan Produksi

Pada penerapan peralatan kerja maka kita harus memperhatikan hal-hal sebagai berikut :

- Spesifikasi teknik
- Kondisi Operasional
- Kalibrasi dan Sistem Pemeliharaan



### II.3. TAHAPAN PROSES PRODUKSI

Pada setiap pembangunan kapal baru selalu melalui proses produksi. Proses pembangunan kapal itu sendiri dimulai sejak material datang sampai dengan penyerahan kapal kepada pihak pemesan (*delivery*). Penerapan proses produksi kapal meliputi beberapa tahap diantaranya yaitu : *Persiapan Produksi* (yang meliputi perancangan dan persiapan gambar kerja, penyimpanan dan pemeriksaan material, persiapan tenaga kerja dan material, dll ), *Fabrikasi* (pemotongan dan pembentukan material), *Sub-assembly* (penggabungan beberapa komponen dasar), *Assembly* (penggabungan beberapa komponen dasar menjadi blok utama), dan *Erection* (penggabungan blok-blok utama menjadi kapal secara utuh).

Pada masing - masing tahap proses produksi tersebut selama pelaksanaannya sering terjadi penyimpangan proses produksi sehingga hal ini akan menghambat jalannya proses produksi karena waktu yang diperlukan untuk pembangunan kapal akan bertambah (*rework*), hal ini tentu akan menambah biaya produksi.



### **II.3.1. TAHAP PERSIAPAN PRODUKSI**

Pada galangan yang akan melaksanakan pembangunan kapal, tahap pertama yang harus dilaksanakan adalah tahap persiapan produksi. Dalam tahap ini mempunyai tujuan mengatur keadaan - keadaan sehingga pada waktu yang ditentukan pekerjaan pembangunan kapal dapat dilaksanakan dan ditetapkan. Pada tahap ini ruang lingkup yang dikerjakan adalah :

- Dokumen produksi ( umum ), yang meliputi ; gambar dan daftar material ; perkiraan kebutuhan tenaga kerja ; perkiraan kebutuhan material.
- Tenaga kerja yang terlibat dalam kaitannya dengan kualifikasi dan jumlah tenaga kerja dan pekerjaan yang lain.
- Material yang perlu dipersiapkan dengan mempertimbangkan ; keadaan atau stock pada gudang ; pemakaian material untuk pekerjaan sekarang ; pemesanan / pembelian material dari luar ( jumlah, waktu pemberian ).
- Fasilitas dan sarana produksi meliputi : kemampuan bengkel produksi ; kapasitas mesin-mesin ; alat-alat angkat yang tersedia ( jumlah, kapasitas, macam dan tempat ) ; keadaan building berth / floating dock ( jumlah, kapasitas dan macam ).

### **II.3.2. TAHAP FABRIKASI**

Pada proses fabrikasi ini, yang penulis amati adalah proses fabrikasi yang terjadi pada Departemen Fabrikasi Lambung dalam Divisi Kapal Niaga di P.T PAL INDONESIA. Departemen ini bertugas membuat komponen kapal dari yang kecil hingga yang terbesar. Jadi bengkel ini merupakan bengkel

awal bagi pembuatan kapal secara fisik. Pada Departemen ini, gambar-gambar dan rambu-rambu yang diperlukan adalah sebagai berikut :

### ***Marking List***

Untuk mengetahui bentuk dari komponen-komponen yang akan dikerjakan dalam satu blok. *Marking List* ini memuat antara lain : nomor kapal, nomor blok, serta ukuran-ukuran dan tanda-tandanya.

### ***Material List***

Untuk mengetahui jumlah dari komponen-komponen yang akan dikerjakan dalam satu blok, berat blok tersebut, dan tempat komponen-komponen tersebut dikerjakan. Selain itu juga untuk mengecek komponen-komponen yang belum dikerjakan.

### ***Cutting Plan***

Untuk mengetahui jumlah dari material-material yang dibutuhkan dalam satu blok baik berupa pelat, profil, bar maupun material yang lain. Pada *cutting plan* ini tercantum pula gambar perencanaan pasisi penggunaan komponen-komponen pelat yang akan dipotong.

### ***Working Drawing***

Adalah gambar bentuk dari blok-blok yang sebenarnya. Untuk bengkel *Fabrikasi* hanya digunakan untuk mengecek kemungkinan terjadi adanya penyimpangan-penyimpangan setelah komponen selesai dan telah dikirim ke bengkel *Assembly*.

### ***Lift dan Scaffold Piece***

Digunakan untuk membuat cincin-cincin ( kupingan-kupingan ) yang digunakan untuk mengangkat blok-blok yang sudah selesai diassembly



untuk diturunkan ke dok. Adapun bentuk dan ukurannya tergantung pada berat blok yang akan diangkat.

Secara prinsip proses produksi pada bengkel Fabrikasi dapat digambarkan sebagai berikut :



Gambar 2.3 Proses Produksi Bengkel Fabrikasi

Keterangan :

#### *Proses Pembersihan*

Sebelum proses pembersihan ini dilakukan maka terlebih dahulu diadakan identifikasi material yang merupakan suatu tindakan pemeriksaan material yang akan dipakai meliputi:

- *Charge Number Material*
- Klasifikasi Material
- Dimensi Material
- Kondisi Permukaan Material

Setelah itu hasil pemeriksaan material dicatat dalam suatu laporan pemeriksaan (*Cheek Sheet*) sekaligus perbaikannya apabila material tersebut mengalami cacat. Pada pelat yang bergelombang harus diperbaiki dengan pelurusan, dengan menggunakan *roll machine*. Pelat tersebut dilewatkan pada suatu susunan silinder baja. Setelah itu baru dimasukkan dalam *short blasting* dan *primary painting machine*. Alat ini berguna untuk menghilangkan karat dan kotoran yang melekat pada material dengan menggunakan butiran-butiran baja yang berdiameter  $\pm 0,5 - 0,8$  mm yang disemprotkan dengan tekanan tinggi, kemudian dicat dengan menggunakan cat dasar.

### *Proses Marking*

Proses *Marking* yaitu proses penandaan pada permukaan material yang akan mengalami pekerjaan sesuai dengan ketentuan tanda kerjanya. Pemindahan data *Mould Loft* serta pengukuran-pengukurannya harus dilakukan seakurat mungkin, karena kesalahan pada proses *marking* ini tidak hanya menyebabkan material yang dipersiapkan untuk *assembly* dan komponen-komponen rusak tetapi juga akan menaikkan jumlah material yang terbuang dan pada akhirnya akan mengurangi nilai produktifitas galangan. Langkah-langkah pekerjaan *marking* adalah :

- Pelat diletakkan diatas lantai yang rata, dan dicek apakah material tersebut sesuai dengan yang ada didaftar.
- Rencana Pemotongan ( *cutting Plan* ), dan pada gambar rencana pemotongan ini diusahakan sisa material sekecil mungkin.

- Pembersihan Material, sebelum di *marking* pelat dibersihkan terlebih dahulu agar kapur dapat melekat betul.
- Persiapan alat - alat kerja.
- Penandaan / Pemberian Nama pada Material

Pada setiap bagian dari material yang telah ditandai harus diberi nama yang jelas agar nantinya tidak tertukar ataupun keliru pada saat perakitan. Nama tersebut disesuaikan dengan kode yang tercantum pada *material list* atau *marking list*. Nama Material terdiri dari : Nomor kapal, Nama kapal, Nama komponen dan posisinya ( *Port side* atau *Starboard side*), Posisi marking ( *up* atau *low marking* )

#### *Proses Cutting ( Pemotongan )*

Adalah proses pemotongan material-material yang telah dimarking dengan cacatan *marking* tersebut telah disetujui oleh *Quality Assurance*.

Alat-alat Pemotongan :

- *Manual Gas Cutting*
- *Semi Automatic Gas Cutting ( Scattor )*
- *NC Gas Cutting Machine*
- *Flame Planner Cutting Machine*

#### *Proses Forming ( Pembentukan )*

Pada proses *Forming* ini dibedakan antara pelat dan profil, Pada pembentukan material pelat ada dua cara yang dilakukan yaitu :



- *Heat Forming ( fairing )*

Cara ini lebih banyak digunakan untuk bentuk-bentuk tiga dimensi atau sebagai penyempurnaan bentuk dari pelat yang telah dibending dengan mesin. Pada prinsipnya cara ini adalah memanaskan pelat kemudian mendinginkan secara mendadak.

- *Cold Forming*

Cara ini dengan menggunakan *bending machine*, biasanya untuk bentuk-bentuk yang sederhana, seperti : *Roll Bending Machine* dan *Horisontal Bending Machine*

Sedang untuk Profil, langkah - langkah yang diperlukan untuk pembentukan adalah :

- Persiapan material antara lain :Profil yang akan *dibending / difairing*, Rambu Film, landasan untuk *fairing*, *Jig* dan lain-lain. Harus diperhatikan gambar-gambar yang ada di rambu film dan Gambar ditempatkan dengan kedudukan yang terbalik.
- Dilakukan penitikan pada tempat-tempat tertentu sesuai dengan rambu film, titik-titik tersebut dihubungkan dengan menggunakan *stroklat* kayu dan *sumitsasi* atau *sumitsubo*.
- Dilakukan perbandingan sedikit demi sedikit dan dilakukan pengecekan setiap kali perbandingan.

### **II.3.3. TAHAP SUB ASSEMBLY**

Proses *sub Assembly* ini merupakan proses kelanjutan dari bengkel *fabrikasi*. Pekerjaan pada bengkel ini meliputi :

- Penyambungan pelat
- Pemasangan *Stifener*
- Merakit *Floor*
- Pemasangan *Face Plate*
- Merakit *Web Frame*



Dalam proses pekerjaan data-data yang diperlukan antara lain : *Yard plan, Working drawing, Material list, Cutting plan, Marking list*

#### **II.3.4. TAHAP ASSEMBLY**

Pada proses *assembly* ini pekerjaan yang dilakukan adalah merakit panel-panel datar, panel lengkap hingga menjadi panel datar, seksi lambung, seksi sekat dan sebagainya sehingga menjadi suatu blok.

##### **Perakitan Panel**

Pada perakitan panel ini, plat-plat diletakkan pada *laticce floor* diatur dan dilas ikat. Urutan pengelasan ikat maupun pengelasannya dimulai dari tengah-tengah panel kemudian secara bertahap keluar, hal ini dilakukan untuk mengurangi deformasi.

Untuk panel-panel yang dibuat dari sambungan banyak pelat dilakukan dengan jalan memberikan beban berat untuk menekan pelat yang akan dilas supaya deformasi yang terjadi sekecil mungkin.

Setelah pelat dilas menjadi satu, pelat diletakkan di atas meja *jig* yang telah disediakan. *Jig* itu harus dicek *levelnya* maupun *countur* dari panel,

setelah itu dilakukan juga pemeriksaan posisi *reference line*, terutama untuk *centre line*, *waterline*, dan *framelinenya*.

Selanjutnya profil-profil dipasang pada garis-garis marking yang telah ditentukan. Profil ini kemudian dilas pada panel.

### **Perakitan blok**

Perakitan blok ini merupakan kelanjutan dari perakitan panel-panel. Di sini perlu diperhatikan pengaturan letak atau posisi dari seksi-seksi sehingga dapat mengurangi pengelasan yang sulit. Pada umumnya bagian yang datar diletakkan pada bagian bawah.

### **II.3.5. TAHAP ERECTION**

Pada proses *erection* adalah kelanjutan dari proses-proses sebelumnya yaitu proses *sub assembly* dan *assembly*. Jenis pekerjaan yang dilakukan pada proses *erection* adalah :

#### ***Loading***

Cara kerja :

- Blok yang ada di pelataran kerja diangkat dengan *crane* yang disesuaikan kapasitasnya.
- Blok yang diangkat harus seimbang dengan meletakkan *ballast* berupa batu cor dengan berat tertentu dan tempat tertentu pula.
- Letak dan besar kupingan harus diperhitungkan.
- Blok ditempatkan pada *keel blok* dan *side blok* yang telah diatur sesuai *marking dock*.



### ***Adjusting***

Cara kerja :

- Blok yang baru harus ditempatkan atau diluruskan *center line* dengan blok yang lama.
- Blok yang baru ditempelkan dikerjakan sesuai dengan ukuran yang telah ditentukan setelah *fitting* dan *welding*.
- Mengatur paju pada *keel block* bila kurang tepat atau rapat menempel pada badan kapal atau block agar tidak terjadi gerakan.

### ***Fitting***

Cara kerja:

- Bagian *center line* harus lurus antara blok yang satu dengan blok yang lain.
- Melihat bagian sisi-sisi kapal apakah sudah tepat dan sesuai dengan ukuran. Bila tidak, maka dilakukan pengukuran jarak dari *center line* ke sisi badan kapal sesuai dengan gambar kerja. Bila terlalu lebar, maka *frame* dibongkar dan *fairing* atau *bending* menurut gambar kerja
- Bila sudah tepat maka ujung-ujung pelat dibentuk kampuh yang sesuai tebal pelat atau gambar kerja.
- Kemudian bila sudah dibuat kampuh las diperiksa lagi kelurusan *center line*, bagian pelat dasar, pelat sisi dan pelat geladak.
- Lalu dilakukan las ikat di tempat-tempat tertentu menurut gambar kerja.

### ***Welding***

Cara kerja :

- Setelah di *fitting* dilakukan pemeriksaan oleh QA dan Class

- Bila test QA atau *Class* disetujui maka dilakukan pengelasan (*welding*) untuk menyambung blok-blok tersebut tentunya dengan metode dan urutan pengelasan yang sesuai dan benar.
- Selain mengelas pelat sisi, pelat dasar, dan pelat geladak juga dilakukan pengelasan atau penyambungan profil-profil pembujur yang dimulai dari *center line* lalu menepi.

### ***Fairing***

Cara kerja :

- Dinding yang cembung dipanasi dengan *brander* las sampai dengan  $\pm 600^{\circ}\text{C}$ .
- Bagian dinding yang cekung disemprot dengan air untuk menarik struktur material akibat pengerutan material tersebut.
- Pemanasan dilakukan merata berupa titik-titik diseluruh dinding yang mengalami deformasi tersebut diatas.

## **II.4. PENGAWASAN PRODUKSI KAPAL**

Pada dekade terakhir ini perkembangan teknologi produksi di bidang pembangunan kapal relatif cukup cepat. Salah satu bentuk teknologi produksi yang mendapat perhatian cukup serius adalah peningkatan pengawasan dan pengendalian mutu di galangan kapal.

Pada setiap proses produksi kapal di suatu galangan, pelaksanaan pengawasan produksi sangat berpengaruh besar terhadap hasil pembangunan kapal.



Fungsi pengawasan terhadap kualitas produksi yang terbatas pada usaha pemenuhan kualitas berdasarkan standar klasifikasi dan standar pemesan. Disamping kedua standar tersebut diatas, diperlukan pula 'standar galangan' yang baru dapat dicapai apabila ditunjang dengan suatu standar kualitas galangan. Standar kualitas diperlukan sebagai tindakan preventif untuk menghindarkan dari terjadinya kealahan atau pekerjaan ulang pada setiap proses pekerjaan.

Tujuan akhir dari dilaksanakannya pengawasan produksi ini adalah untuk menekan biaya produksi dan meningkatkan mutu kapal hasil pembangunan suatu galangan.

Pengawasan haruslah dilakukan sedini mungkin yakni dimulai dari desain, produksi sampai dengan *performance* hasil produksi tersebut. Sehingga penyimpangan –penyimpangan dari standar kualitas maupun spesifikasi dapat dihindari sedari awal dan apabila dialami atau ditemui penyimpangan-penyimpangan dapat diatasi sesuai dengan prosedur yang diijinkan. Dengan demikian biaya produksi dan waktu produksi dapat ditekan atau ditepati dan yang tidak kalah pentingnya adalah kualitas produksi dapat dijamin/ dipertanggung jawabkan.

Dalam pelaksanaan pengawasan produksi, pengawasan dan pemeriksaan ketepatan dilakukan tiap hari menurut jadual yang telah ditentukan yaitu: jadual pembangunan kapal, dengan kegiatan pokok mengadakan pemeriksaan, pengukuran, dan pencatatan data-data hasil pengukuran.

Pemeriksaan secara langsung atau *visual* mengenai mutu hasil pekerjaan pada setiap proses produksi dilakukan secara intensif oleh



*checker*, QA/QC dan manajer proyek. Sedangkan peranan *surveyor* BKI dan *Owner surveyor* dalam hal pengawasan dan pengendalian mutu adalah memeriksa agar mutu kapal tersebut tidak menyimpang dari standar dan spesifikasi yang telah ditentukan sebelumnya.

Tempat-tempat yang perlu dicek dapat langsung diketahui oleh pekerja melalui lembar *check sheet* yang diterima dari QA/QC. Untuk tiap tahap pekerjaan, item-item yang perlu diperiksa kecocokannya dengan standar yang telah ditentukan sebagai berikut:

**a. Tahap Persiapan,**

meliputi: dimensi-dimensi secara keseluruhan termasuk kelebihan-kelebihan yang diberikan dan tanda-tanda yang dimaksud untuk fabrikasi, perakitan, dan pemeriksaan pekerjaan.

**b. Tahap Fabrikasi**

meliputi: dimensi-dimensi secara keseluruhan dari pelat-pelat atau bagian-bagian yang telah terpotong, persiapan-persiapan permukaan, deformasi, dan kelengkungan dari pada komponen yang melengkung.

**c. Tahap Perakitan ( *sub-assembly dan assembly* )**

meliputi: penyetelan komponen-komponen yang akan dirakit, jarak antara dua bagian yang akan dilas, dan menjaga kelurusan badan kapal.

**d. Tahap Penyambungan ( *Erection* )**

meliputi: penyambungan block-block, gap-gap untuk pengelasan, dan pemeliharaan kelurusan badan kapal.

QA/QC memberikan umpan balik (*feed back*) kepada bengkel Assembly mengenai masalah-masalah yang muncul pada block. Umpan balik tersebut

juga memberikan komentar mengenai titik-titik yang vital dan dimensi-dimensi yang berkaitan dengan proses erection.

Tujuan dari pada umpan balik tersebut adalah untuk memberikan informasi dan komentar mengenai bagaimana dan apa yang seharusnya dilakukan untuk memperbaiki detail-detail disain, toleransi-toleransi, metode-metode kerja, dan toleransi untuk deformasi.

### **Sistim Pemeriksaan Kualitas Lambung**

Sistim pemeriksaan lambung dilaksanakan dengan *Check Sheet System* yang mana pemeriksaan kualitas hasil produksi dengan mempergunakan *check sheet* ( lembar pemeriksaan ) sesuai dengan proses pembangunan kapal.

#### **Pemeriksaan pada kualitas lambung perlu dilakukan karena :**

- Hasil kerja dapat diawasi dan di evaluasi sedari awal sampai dengan selesai oleh Surveyor galangan.
- *Surveyor* Badan Klasifikasi pada beberapa pemeriksaan mungkin cukup hanya memeriksa hasil *check sheet* saja.
- Penyampaian informasi tentang problem teknik dapat diperoleh melalui analisa pada *check sheet* yang dihasilkan.

QA/QC yang beranggotakan pekerja pada bagian disain konstruksi bertugas membuat *check sheet* yang berisi tempat-tempat yang akan dicheck, gambar konstruksi yang akan dicheck, tempat pencatatan data, penyimpangan yang diijinkan, serta waktu pencatatan, *working area*, tahap pekerjaan, identitas pelaksana dan identitas *petugas QA/QC*. *Check sheet*



harus dipersiapkan untuk semua tahapan proses produksi mulai dari fabrikasi sampai dengan erection.

### *Check Sheet untuk Fabrikasi*

Fabrikasi adalah awal tahapan proses pembangunan kapal. Pekerjaan yang dilaksanakan adalah *marking, cutting, dan bending*. Kesalahan atau penyimpangan yang kecil akan mempengaruhi proses selanjutnya. Oleh karena itu diperlukan standar akurasi tertentu. Komponen yang diperiksa dan diukur dipilih secara random oleh bagian *mould loft* dikarenakan material fabrikasi sangatlah banyak jumlahnya. Petunjuk-petunjuk pengukuran ditentukan oleh bagian *mould loft*.

Yang perlu diperhatikan pada waktu pengukuran adalah :

- Derajat kemiringan rambu-rambu.
- Ketepatan sisi pelat dengan garis tanda pada *template*.
- Jarak antara sisi *template* dengan pengukuran pelat.
- Kelengkungan memanjang dan melintang.
- Puntiran dan
- Kelurusan garis pandang sumbu referensi pada rambu.



### *Check Sheet untuk sub-block Assembly*

Yang penting pada *sub-block assembly* untuk *accuracy control* adalah menempatkan *stiffener*, letak *bracket, face plate* dan sebagainya. Bagaimana menghindari deformasi dan penyusutan yang disebabkan oleh pengelasan. Aktivitas pengukuran selama proses *sub-block assembly* adalah:



- Pengecekan dimensi *fitting*
- Pengukuran panjang, lebar, dan diagonal pelat
- Pengecekan untuk deformasi dan penyusutan dengan menggunakan *references line* pada *web plate*
- Pengukuran dimensi-dimensi seperti tercantum pada *check sheet*

Mempertahankan akurasi pada *block assembly* adalah penting karena ini adalah tahap akhir sebelum *block dierection* di *building berth*. Harus dibedakan bentuk pemeriksaan untuk bentuk block lurus dan bentuk block lengkung. Hal-hal yang perlu diperhatikan pada pengukuran adalah:

- Pengukuran panjang, lebar dan diadonal setelah *base panel diassembly*
- Puntiran
- Lokasi dari sub-block dan internal part setelah dipasang pada *base panel*

Untuk bentuk block lengkung hal-hal yang perlu dipertahankan adalah:

- Pengukuran panjang, lebar dan diagonal block
- Pengukuran petunjuk ketepatan lokasi pelat lengkung untuk sebuah *base panel* pada sebuah *pin jig*
- Sarana untuk pemeriksaan
- Pengecekan pada dimensi-dimensi yang dibutuhkan pada tepi-tepi panel yang mana nanti dibutuhkan pada waktu penyambungan *dierection*
- Instruksi-instruksi bagaimana untuk mendeteksi dan mengoreksi deformasi yang disebabkan oleh pengelasan
- Puntiran

### *Check Sheet Untuk Erection*

Selama proses *erection* harus dimonitor dimensi-dimensi block seperti panjang, lebar *Leveling* dan tinggi karena ini menunjang ukuran kapal sesuai yang disetujui pemesan. Data-data harus dicatat serta variasi-variasi yang muncul. Anggota QA/QC secara periodik melakukan pengecekan terutama pada vital-vital point dari mula *keel laying* sampai dengan *launching*.

## BAB III

# PENYIMPANGAN DIMENSI DAN BENTUK PADA PROSES PRODUKSI KAPAL

### III.1. TEORI DASAR STANDAR PENYIMPANGAN DIMENSI DAN BENTUK

Sebagai dasar untuk melaksanakan pemeriksaan diperlukan suatu standar sebagai pedoman, sehingga dari standar ini dapat diketahui berapa besar penyimpangan yang terjadi dan dapat diketahui apakah penyimpangan tersebut masih dalam batas toleransi yang diijinkan atau tidak.

Pengertian standar yaitu suatu proses kerja berupa suatu ketentuan-ketentuan yang digunakan perusahaan untuk menjamin proses kerja berjalan stabil dan hasil kerjanya sesuai dengan yang diinginkan. Salah satu fungsi standarisasi adalah untuk menyeragamkan pelaksanaan pekerjaan pada setiap tahap produksi.

Standar yang ada dan dipergunakan pada galangan kapal adalah yang diperoleh dari *JSQS ( Japanese Shipbuilding Quality Standard )* yang memuat *standard range* dan *batas toleransi* untuk konstruksi lambung.

Pada perencanaan standarisasi kerja, terdapat beberapa hal yang perlu diperhatikan

#### a. Standarisasi *Excess*

- Tahapan, tempat, proses dan besar *excess* yang diperlukan.
- Tahapan, tempat, proses dan besar *margin* diperlukan.

#### b. Standarisasi untuk *Allowance* penyusutan

- Faktor yang disebabkan oleh prosedur pengelasan dan pemotongan .



- Faktor sifat mekanis material terhadap perilaku proses.
- Faktor sistem kerja mesin dan batas ketelitian.

**c. Standarisasi garis dasar dan titik pertemuan**

- Lokasi
- Panjang

**d. Standarisasi produksi**

- Urutan pekerjaan *fabrikasi*
- Urutan pekerjaan *sub-assembly*
- Urutan pekerjaan *assembly*
- Urutan pekerjaan *erection*

**e. Standarisasi prosedur pemeriksaan**

- Prosedur tertulis mengenai metode pengukuran pada setiap proses.
- Sistem pengisian kartu pemeriksaan A/C
- Lembar umpan balik (laporan hasil pengukuran).

**f. Standarisasi informasi A/C**

- Detail struktur konstruksi
- Persiapan pemotongan
- *Allowance* untuk *excess*
- Metode fabrikasi, titik vital dan dimensi

**III.1.1. Penyimpangan Terhadap Bentuk dan Ukuran dari Perencanaannya.**

Penyebab masalah ini terutama karena deformasi dan juga karena kesalahan pemasangan elemen - elemen penyusun konstruksi. Bentuk dari deformasi yang sering dijumpai meliputi :

- Deformasi Penyusutan

- Deformasi Sudut
- Deformasi *Buckling*
- Deformasi Memanjang
- Deformasi lainnya.

#### A. Penyimpangan Terhadap Bentuk dan Ukuran karena Penyusutan.

Penyusutan ini dikarenakan proses pengerjaan maupun karena material yang digunakan yaitu material plat dan material las.

1. Penyusutan karena proses pengerjaan antara lain adalah :

##### **a. Pengerjaan yang dilakukan pada tahap perencanaan dan *mouldloft*.**

Penyusutan disini disebabkan karena peralatan, operator, metode yang digunakan. Hal ini hubungannya adalah terhadap ketelitian dari hasil kerjanya. Pada tahap ini yang berpengaruh terhadap penyusutan adalah : pemilihan jenis rambu dan pemilihan metode pengembangan kulit. Disini penggunaan hal-hal diatas harus disesuaikan dengan keadaan sehingga akan diperoleh hasil yang teliti.

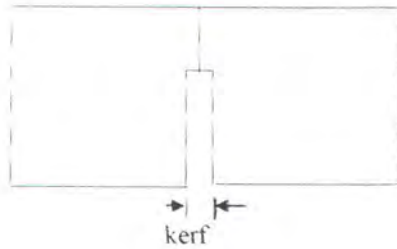
##### **b. Pengerjaan selama Fabrikasi**

Pengerjaan pada tahap ini yang berpengaruh terhadap penyusutan adalah penandaan, pemotongan dan pembentukan.

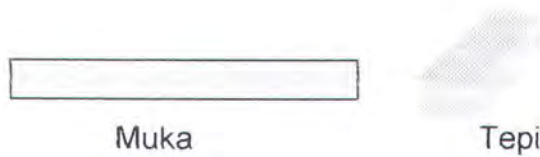
Pengaruh penandaan terhadap penyusutan disebabkan oleh ketelitian operator, penggunaan peralatan dan kode yang digunakan.

Pengaruh pemotongan terhadap penyusutan adalah adanya *kerf*, kertaan permukaan dan distorsi pemotongan.

*Kerf* adalah kehilangan sebagian material karena proses pemotongan. Besar *kerf* tergantung dari tebal plat, alat yang digunakan dan operator.



Kerataan permukaan berpengaruh terhadap penyusutan karena adanya pekerjaan ulang terhadap permukaan potongan dapat dilihat dibawah ini:  
Potongan yang baik adalah seperti gambar dibawah ini.



Penyebab: tinggi *nozzle* dan penyetelan api yang tepat serta kecepatan jalan yang tepat. Ukuran *nozzle* dan tahanan yang tepat untuk plat yang dipotong.

Identifikasi: pinggiran puncak dan dasar berbentuk siku dan tajam. Garis-garis tarikan vertikal yang agak melengkung pada bagian dasar serta tidak terdapat penempelan bahan yang terbuang. Oksida mudah dibersihkan untuk menghasilkan permukaan babas kerutan.

Distorsi pemotongan biasanya disebabkan adanya *internal deformation* karena proses pemanasan yang terpusat, dan juga karena adanya *external deformation* karena adanya penahanan atau penekanan. Pengaruh lain adalah teknik pemotongan, arah pemotongan yaitu pemotongan yang dilakukan dengan salah satu bagian ujung diklem. Pada bagian plat yang tidak diklem deformasi akibat pemotongan menyebabkan plat bergerak



pada penahannya dengan arah gerakan yang terkecil. Untuk mengatasi distorsi ini perlu adanya pemotongan yang benar urutannya serta memberikan bentuk lawan dari terjadinya distorsi.

Pembentukan, pengaruhnya terhadap penyusutan adalah perubahan bentuk banding. Dalam pembentukkan dengan cara panas akan terjadi *internal deformation* seperti pada pemotongan, sehingga penyusutan sering terjadi disini. Selain itu juga perlu diperhatikan masalah kecenderungan plat untuk kembali pada bentuknya semula, karena adanya deformasi elastis.

### **c. Pengerjaan selama sub assembly dan assembly.**

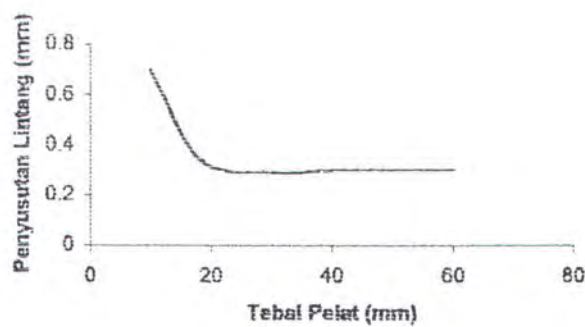
Pengerjaan pada tahap ini yang berpengaruh terhadap penyusutan adalah pada pekerjaan *fitting* dan *welding*.

Pengaruh pekerjaan *fitting* terhadap penyusutan adalah ketelitiannya dalam hal pemasangan elemen maupun detail sebelum dilakukan pengelasan.

Sedangkan pengaruh pengelasan terhadap penyusutan adalah karena adanya distorsi. Distorsi ini disebabkan karena adanya gaya dari dalam dan dari luar.

Hai-hai lain yang mempengaruhi penyudutan adalah:

1. Tebal Pelat

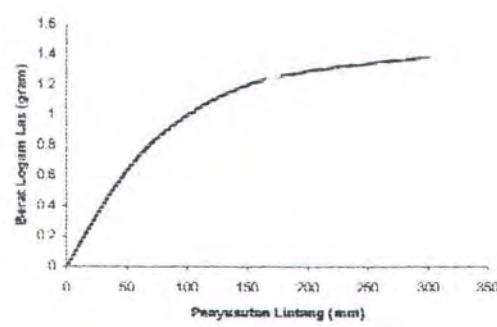


Gambar 3.1 Grafik hubungan pengaruh tebal terhadap penyusutan melintang.

Pada diagram diatas menunjukkan pengaruh tebal plat terhadap penyusutan melintang pada sambungan tumpul.

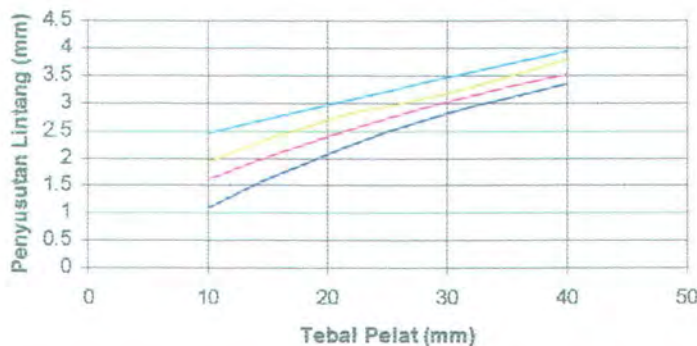
2. Berat Logam Las.

Secara diagram dapat ditunjukkan hubungan antara penyusutan dengan berat logam las.



Gambar 3.2. Grafik Hubungan antara penyusutan dengan berat logam las

### 3. Celah Pengelasan.



Gambar 3.3 Grafik Pengaruh Tebal pelat dan celah akar terhadap penyusutan

Pengaruh tebal plat dan celah akar terhadap penyusutan dapat dilihat pada gambar diatas. Penyusutan pada celah akar dengan alur V akan semakin besar dengan bertambah besarnya tebal plat.

## 2. Penyusutan karena material antara lain meliputi:

### a. Material Plat.

Hal-hal yang berpengaruh terhadap penyusutan adalah :

Keadaan fisik material, yaitu lengkungan, laminasi, cacat dan lain-lain, yang berpengaruh dalam proses *marking*, *cutting* dan *welding*. Keadaan fisik material tersebut mempengaruhi proses *marking* karena ketelitian proses ini sangat tergantung dari keadaan material.

Sifat material, yaitu harga koefisien muai, harga koefisien perambatan panas , kadar karbon dan daya elastisitas baja. Harga koefisien muai semakin tinggi harganya semakin besar kemungkinan terjadinya distorsi yang timbul. Harga ini bertambah besar dengan pertambahan suhu. Sehingga penyusutan yang diakibatkan oleh adanya distorsi juga semakin



besar.

#### **b. Material Las.**

Material las mempunyai pengaruh terhadap penyusutan sambungan, maka dari itu perlu diperhatikan dalam hal:

- Ukuran *elektrode*, dimana diameter yang besar akan memberikan penyusutan yang lebih besar, tetapi hal ini tidak selamanya berlaku.
- Jenis *flux*, dimana jenis *flux* ini akan mempengaruhi kecepatan perpindahan logam las ke base metal, heat input, dan lain-lain. Dengan adanya perbedaan ini akan menyebabkan perbedaan pula terhadap timbulnya penyusutan.

#### ***B. Penyimpangan terhadap bentuk dan ukuran karena deformasi sudut.***

Deformasi sudut dapat terjadi apabila pada dua permukaan material terdapat perbedaan penyusutan atau pengembangan. Deformasi sudut akan membuat konstruksi yang direncanakan berubah dari bentuk dan ukurannya. Yang dimaksud dengan perubahan bentuk dan ukuran disini adalah perubahan sudut pada sambungan las.

Perubahan sudut ini disebabkan karena adanya proses pengerjaan dan material yang digunakan.

#### **1. Deformasi sudut karena proses pengerjaan.**

Proses pengerjaan yang mempengaruhi terjadinya perubahan sudut adalah:

##### **a. Fitting.**

Proses *Fitting* ini dilakukan pada tahap *sub-assembly* dan *Assembly*. yang sangat berpengaruh adalah dalam hal ketelitiannya waktu

melaksanakan pemasangan elemen-elemen atau detail pada tempatnya. Kesulitannya adalah karena kita harus memperkirakan seberapa besar perubahan sudut setelah dilakukan pengelasan.

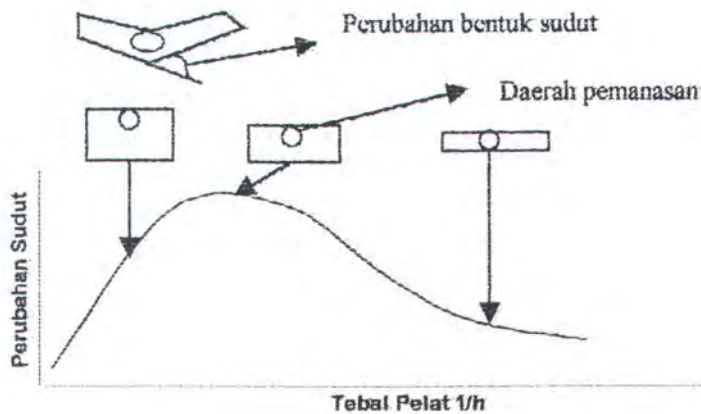
### **b. Pengelasan.**

Proses pengelasan ini sekarang tidak dapat dihindari lagi, karena penyambungan dengan las adalah yang paling efisien, namun tidak berarti tanpa kerugian. Kerugian ini antara lain menyebabkan sambungan las berubah sudutnya dari yang direncanakan.

Proses pengelasan ini erat kaitannya dengan pekerjaan fitting dan sekaligus memperkirakan perubahan sudut yang akan terjadi.

Hal-hal lain yang berpengaruh terhadap perubahan sudut antara lain:

#### **1. Tebal Pelat.**

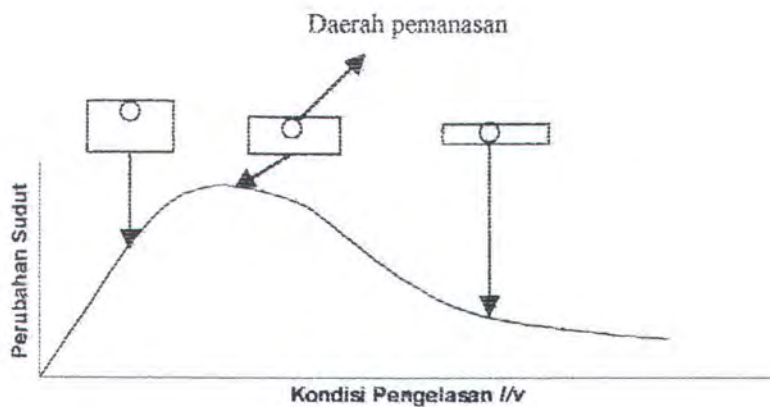


Gambar 3.4 Grafik Hubungan antara Perubahan Sudut dengan Tebal Pelat

Diagram skematik hubungan antara perubahan sudut dengan tebal pelat pada kondisi las yang tetap dapat dilihat seperti diatas. Pada pengelasan pelat yang tipis perubahan sudut yang terjadi kecil, hal ini disebabkan karena bentuk *weld metal* mendekati simetri. Dengan semakin

tebal pelat, maka deformasi sudut yang terjadi semakin besar, karena bentuk *weld metal* menjauhi bentuk simetri sehingga penyusutan atau pengembangan antara dua permukaan material perbedaannya semakin jauh. Tetapi pada ketebalan pelat tertentu deformasi sudut yang terjadi semakin kecil, hal ini dikarenakan adanya penahanan dari pelat yang semakin besar dengan pertambahan ketebalan pelat.

## 2. Kondisi Pengelasan.



Gambar 3.5 Grafik Kondisi Pengelasan

Kondisi pengelasan yang dimaksud meliputi arus pengelasan, kecepatan pengelasan dan tebal pelat.

Pada diagram diatas ditunjukkan hubungan antara kondisi pengelasan dalam batang uji manik dengan diameter elektrode 4 mm dan 6 mm.

Arus dari pengelasan dan kecepatan pengelasan pengaruhnya terhadap perubahan sudut adalah pada jumlah atau bentuk dari *weld metal* yang besar akan membuat bentuk grove mendekati simetri, sehingga perubahan sudut yang terjadi semakin besar. Pada kondisi tertentu



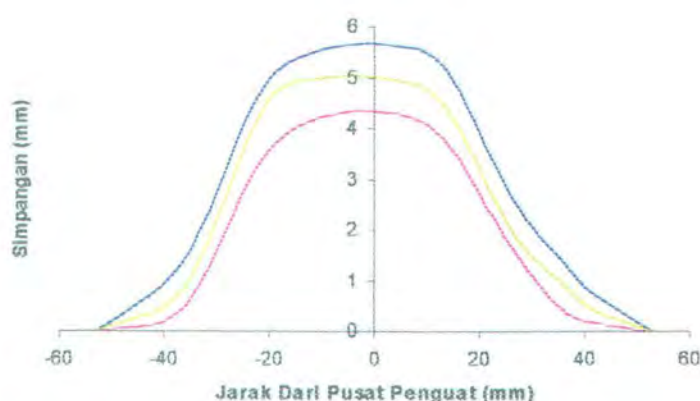
perubahan sudut yang terjadi semakin besar. Pada kondisi tertentu perubahan sudut ini akan mengecil, hal ini karena adanya penahan dari pelat yang semakin besar dengan makin bertambah tebalnya pelat.

### 3. Bentuk Alur.

Bentuk alur dipengaruhi apakah alur berbentuk V atau alur bentuk X. Pada bentuk V akan terjadi perubahan bentuk yang besar, hal ini disebabkan karena adanya penyusutan dari weld metal yang besarnya tidak sama pada dua permukaan material.

Sedangkan pada alur X yang simetri perubahan sudut yang terjadi kecil, hal ini disebabkan karena perbedaan penyusutan weld metal pada dua permukaan material hampir sama. Pada bentuk alur yang semakin tidak simetri akan terjadi perbedaan penyusutan yang semakin besar sehingga perubahan sudut yang terjadi juga semakin besar.

### 4. Jumlah Lapisan.

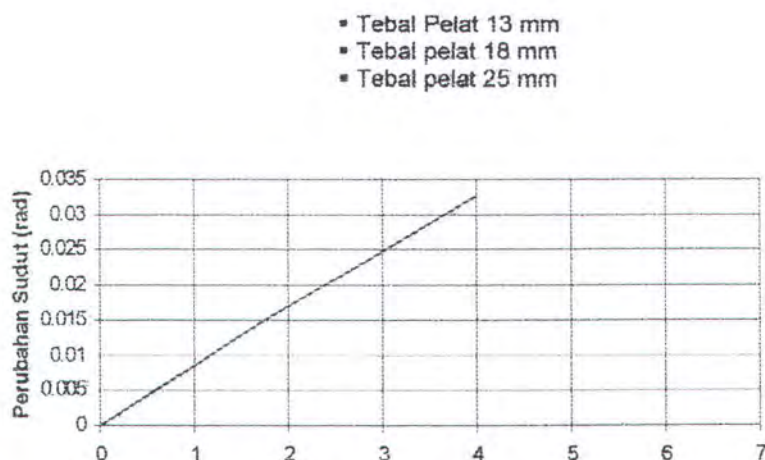


Gambar 3.6 Grafik Hubungan antara Jumlah Pengelasan Terhadap Penyimpangan pada Sambungan T

Diagram diatas menunjukkan pengaruh jumlah lapisan terhadap penyimpangan pada sambungan T. Besarnya perubahan sudut pada

sambungan T akan semakin besar dengan semakin banyaknya jumlah *pass*. Hal ini dikarenakan dengan semakin banyaknya *pass*, maka semakin besar pula gaya yang timbul akibat penyusutan dari *weld metal*. Pada jarak yang semakin jauh dari pengelasan, perubahan sudut (*rad*) akan semakin kecil karena pengaruh dari *weld metal* semakin kecil.

## 5. Berat logam Las.

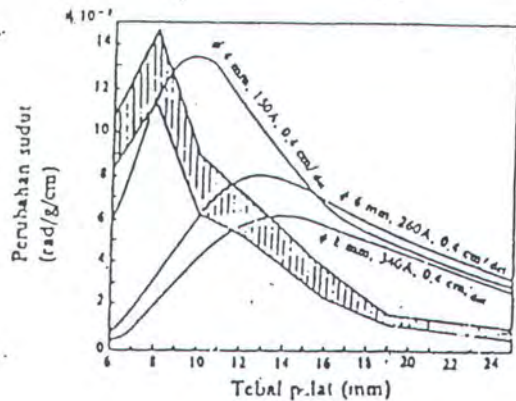


Gambar 3.7 Grafik Hubungan antara Berat Logam Las dengan Perubahan Sudut

Hubungan antara berat logam las dengan perubahan sudut dapat dilihat pada diagram diatas. Dengan semakin beratnya logam las, maka perubahan sudut (*rad*) akan semakin besar dan semakin besar atau beratnya logam las, maka besarnya gaya penyusutan yang mengakibatkan terjadinya perubahan sudut semakin besar.

## 6. Prosedur Pengelasan.

Prosedur pengelasan yang dimaksud adalah kondisi pengelasan dan tebal plat yang digunakan. Secara diagram hubungan antara prosedur pengelasan dan perubahan sudut dapat dilihat pada gambar dibawah.



Gambar 3.8 Grafik Hubungan antara Prosedur Pengelasan dan Perubahan Wujud

Diagram diatas dapat digunakan untuk memilih metode pengelasan yang sesuai agar perubahan sudut yang terjadi seminim mungkin. Dari diagram tersebut dapat dilihat bahwa bila tebal plat kurang dari 10 mm, las busur listrik tangan dengan elektrode yang besar memberikan perubahan sudut yang lebih kecil daripada yang terjadi karena busur redam. Tetapi bila plat yang dilas tebalnya lebih dari 12 mm, maka hal sebaliknya yang akan terjadi.

### C. Penyimpangan Terhadap Bentuk dan ukuran Karena Deformasi

#### *Memanjang.*

#### **1. Deformasi Memanjang karena Proses Pemotongan.**

Pada pekerjaan pemotongan ini material akan mengalami deformasi bila prosedur pemotongan yang digunakan tidak tepat. Proses pemotongan



dapat dilakukan secara mekanis maupun thermal.

Secara mekanis adalah dengan menggunakan gunting, sehingga pada ujung-ujung pelat yang dekat daerah pemotongan akan timbul deformasi. Deformasi ini sifatnya hanya kecil dan biasanya diperbaiki dengan pengerolan atau dengan press. Pemotongan secara thermal terhadap material pelat merupakan cara yang paling banyak dilakukan. Pemotongan ini dilakukan pada tahap *fabrikasi*, *sub assembly*, *assembly* maupun pada tahap *erection*. Pemotongan *thermal* ini akan menyebabkan material menerima suhu yang tidak merata pada material secara keseluruhan. Karena distribusi panas yang tidak merata ini akan timbul tegangan dan perubahan bentuk dari material yang dipotong.

Sifat deformasi karena proses pemotongan pada bermacam-macam ketebalan pelat adalah sama, yaitu dengan semakin lebar pelat yang dipotong maka deformasi yang ditimbulkan akan semakin kecil, hal ini dikarenakan adanya penahanan dari material yang semakin besar.

## **2. Deformasi Memanjang karena Proses Pengelasan.**

Seperti pada pemotongan dengan panas, pada proses pengelasan juga ditimbulkan panas yang terpusat pada material. Karena adanya distribusi panas yang tidak merata, terutama pada dua permukaan material yang dilas, maka akan timbul pemuaian material yang berbeda pula antara dua permukaan material tersebut. Demikian juga dengan proses pendinginan.

## **3. Deformasi Memanjang karena Proses Penanganan.**

Penanganan harus dilakukan dengan hati-hati dan benar, penanganan

proses ini sangat berpengaruh terhadap benda yang ditangani (diangkat) terutama terhadap bentuknya. Pada penanganan yang salah sering mengakibatkan terjadinya perubahan bentuk (deformasi) pada konstruksinya.

Terutama pada pengangkutan bentuk seksi-seksi atau block-block yang berat, letak *lifting piece* harus benar. Terjadinya deformasi ini disebabkan karena adanya gaya luar, yaitu gaya berat dari konstruksi tersebut, gaya tarik (gaya dari proses pengangkutan), gaya geser torsi dan lain-lain.

#### **4. Faktor-faktor yang Mempengaruhi Terjadinya Deformasi.**

##### **a. Panas yang Masuk pada Material.**

Bila panas yang masuk pada material semakin banyak, maka deformasi yang terjadi cenderung semakin besar, besarnya panas ini dipengaruhi oleh :Arus pengelasan, pemanasan mula, kecepatan pengelasan, beda tegangan, diameter elektrode dan lain-lain.

##### **b. Harga Koefisien Muai.**

Harga koefisien muai semakin besar maka pemuaian atau penyusutan dari material karena proses thermal akan semakin besar, sehingga perubahan bentuk atau deformasi yang terjadi cenderung semakin besar juga.

##### **c. Harga Koefisien Perambatan Panas.**

Harga koefisien perambatan panas semakin besar maka deformasi yang terjadi akan semakin kecil, hal ini disebabkan karena panas pada material akan mudah didistribusikan secara merata, sehingga peregangan yang terjadi relatif lebih kecil. Dengan adanya peregangan yang kecil ini

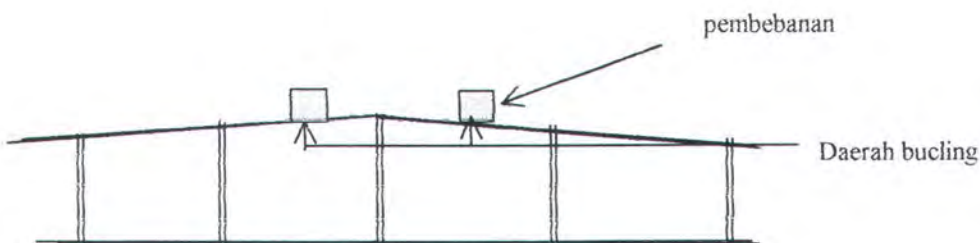


maka deformasi yang terjadi juga semakin minim.

#### D. Penyimpangan Bentuk dan Ukuran Karena Deformasi *Buckling*.

Masalah *buckling* ini dapat terjadi karena adanya gaya dalam maupun karena adanya gaya luar. Gaya dalam yang berpengaruh terhadap *buckling* adalah gaya-gaya yang timbul akibat proses pengelasan maupun proses pemotongan thermal.

Dengan adanya proses pemotongan dan pengelasan ini maka pada material dapat timbul tegangan sisa. Tegangan sisa pada konstruksi akan mempermudah terjadinya deformasi *buckling* apabila pada konstruksi tersebut menerima beban. Hal ini akan semakin cenderung terjadinya *buckling* bila pengelasan dilakukan pada pelat yang panjangnya melampaui panjang yang ditentukan untuk tebal pelat yang diberikan. Sedangkan gaya luar yang sering menyebabkan deformasi *buckling* ini adalah adanya beban yang berlebihan. Pembenahan ini sering kali dilakukan pada saat proses pengelasan pada bagian-bagian yang panjang, karena pembebanan atau penahanan ini bertujuan untuk menghindari terjadinya deformasi memanjang pada pelat yang dilas.



Gambar 3.9 Pembebanan Pada Proses Pengelasan



Beban yang lain dapat juga terjadi karena adanya konstruksi yang berbeda diatasnya. *Buckling* ini mempunyai bentuk fisik yang lebih dari satu tempat permukaan. Kecenderungan terjadinya *buckling* ini biasanya pada pelat yang tipis yang menerima baban tekan dan telah mendapat pengerjaan pengelasan.

#### **E. Penyimpangan Terhadap Bentuk dan Ukuran Karena Kesalahan Pemasangan.**

Perubahan bentuk pada bagian-bagian konstruksi yang disebabkan karena kesalahan pemasangan ini masih sering kita jumpai didalam pembangunan kapal. Kesalahan ini biasanya dikarenakan kurang telitian pada saat pemasangan bagian-bagian konstruksi sebelum dilakukan pengelasan, sehingga konstruksi yang dihasilkannya telah berubah dari yang direncanakan.

Adanya kesalah pahaman pada saat membaca gambar sehingga akan merubah pada konstruksi yang dibuat. Penyebab lain dalam hal ini adalah karena kecerobohan dalam pemasangan bagian-bagian konstruksi sehingga terjadi kekeliruan penempatan bagian-bagian konstruksi yang dipasang.

Dengan adanya kesalahan pemasangan ini menyebabkan bentuk dan ukuran konstruksi menjadi berubah dari yang direncanakan. Hal ini dapat menimbulkan masalah-masalah pada tahap erection nantinya karena akan timbul ketidak tepatan pada bagian-bagian yang akan disambung.

### III.2. JENIS PENYIMPANGAN DIMENSI DAN BENTUK PADA PROSES PRODUKSI KAPAL

Selama proses produksi pada pembangunan kapal sedang berjalan terdapat beberapa permasalahan yang sering timbul pada tiap tahapan proses produksi, yaitu :

#### 1. PROSES *FABRIKASI*

##### ***Marking***

Kesalahan pada umumnya disebabkan kesalahan pemberian tanda atau nama, yang terdiri dari :

- tanda pengerjaan
- tanda urutan dan arah pengerjaan
- tanda lokasi komponen
- tanda jenis dan macam komponen
- tanda ukuran dan dimensi komponen

##### ***Cutting***

Macam kesalahan yang terjadi :

- Kekasaran permukaan potongan
- Penyusutan material karena panas saat pemotongan

##### ***Bending***

Macam kesalahan yang terjadi :

- Kesalahan pencocokan rambu bending
- Kesalahan sudut bending
- Pelat berubah bentuk dengan sendirinya setelah selesai dibending, baik

dengan proses dingin maupun panas.

## 2. PROSES SUB – ASSEMBLY DAN ASSEMBLY

### ***Fitting***

Ketidak telitian akurasi dimensi struktur atau komponen pada pekerjaan *fitting*, umumnya dari kejadian-kejadian berikut :

- *Misalignment* atau ketidaklurusan bagian terpasang.
- Gap atau celah yaitu jarak antara dua bagian yang akan disambung.
- *Misfitting*, yaitu kesalahan tempat pemasangan elemen-elemen pada detailnya.
- Penyimpangan sudut pemasangan antara profil dengan pelat maupun dengan profilnya sendiri.

### ***Welding***

Pada proses *welding* akibat perlakuan panas pengelasan pada material umumnya ialah :

- Penyusutan memanjang
- Penyusutan melintang
- *Angular distortion*, pengaruhnya pada penyimpangan sudut pada *fillet weld*.

### ***Marking akhir***

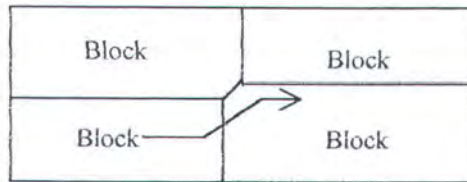
Kesalahan penandaan meliputi tanda-tanda : posisi *fitting*, nama bagian, letak pada konstruksi *hull*, sudut-sudut *fitting* dan proses pekerjaan akhir



### 3. PROSES ERECTION

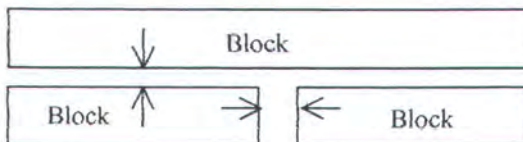
Macam dari penyimpangan terhadap bentuk dan ukuran dari perencanaannya yang dijumpai pada tahap erection antara lain adalah:

- ***Terjadinya ketidak lurusan pada pertemuan sambungan antara block-block atau seksi-seksi.***



Penyebabnya adalah karena adanya penyusutan pada block atau seksi pada saat proses pembuatannya.

- ***Terjadinya gap-gap pada sambungan tumpul diantara block-block atau seksi-seksi.***



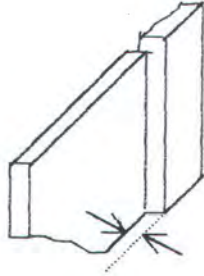
Penyebabnya adalah karena adanya penyusutan pada waktu proses pembuatan block atau seksi.

- ***Terjadinya gap pada sambungan T antara hubungan sekat melintang dengan pelat geladak.***



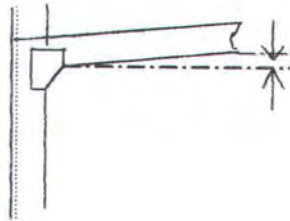
Penyebabnya adalah karena adanya penyusutan pada saat pembuatan sekat melintang.

- ***Terjadinya ketidak lurusan (misalignment) pada konstruksi sekat memanjang.***



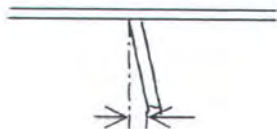
Penyebabnya adalah karena adanya kesalahan pemasangan, deformasi dan perubahan sudut pada sambungan pengelasannya.

- ***Terjadinya perubahan sudut antara konstruksi lambung dan geladak.***



Penyebabnya adalah karena adanya perubahan sudut, penyusutan dan deformasi memanjang.

- ***Terjadinya perubahan sudut antara pelat alas dalam dan pelat lunas.***

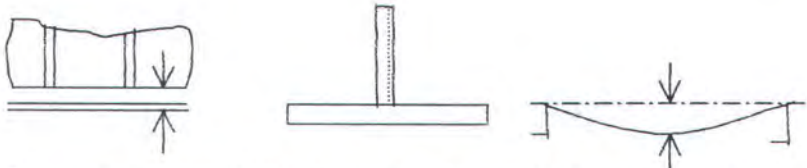


Penyebabnya adalah karena adanya deformasi sudut atau kesalahan pemasangan.

- ***Terjadinya penyimpangan sudut antara pelat alas dalam dan sekat melintang.***

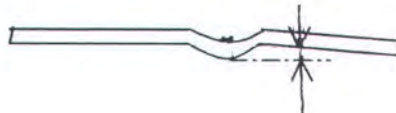
Penyebabnya adalah karena adanya kesalahan pemasangan, deformasi memanjang, deformasi sudut.

- ***Terjadinya penyimpangan pelat diantara titik tumpuannya.***



Penyebabnya adalah karena adanya deformasi sudut.

- ***Terjadinya angular misalignment pada sambungan tumpul, yaitu pada pelat kulit, pelat alas dalam, pelat sekat dan lain-lain.***



Penyebabnya adalah karena adanya deformasi sudut.

- ***Terjadinya penyimpangan dari garis lurus pada pelat kulit, pelat geladak dan lain-lain.***



Penyebabnya adalah karena adanya deformasi sudut.



- ***Terjadinya ketidak tepatan pada sambungan pengelasan (seamslag) pada pelat alas dalam, pelat sekat dan lain-lain.***

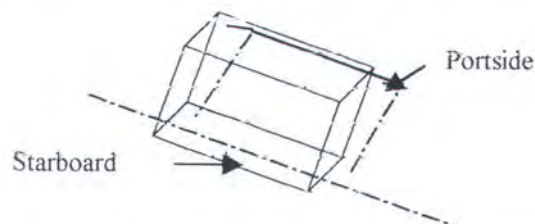


Penyebabnya adalah karena kurang telitian pemasangan, perencanaan dan deformasi.

- ***Terjadinya penyimpangan kearah kelebaran dari beban kapal.***

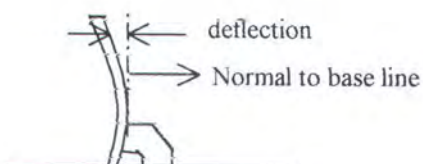
Penyebabnya adalah karena adanya penyusutan, deformasi sudut, deformasi memanjang dan kesalahan pemasangan.

- ***Terjadinya distorsi selatif pada block lambung.***



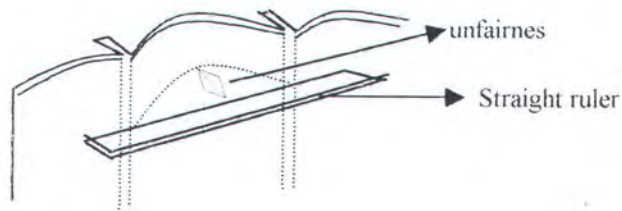
Penyebabnya adalah karena adanya deformasi sudut.

- ***Terjadinya defleksi pada block lambung.***



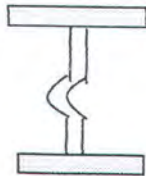
Penyebabnya adalah karena adanya deformasi memanjang.

- ***Terjadinya lekuk pada pelat kulit.***



Penyebabnya adalah karena adanya kesalahan atau kurang telitian ukuran, penyusutan.

- ***Terjadinya lekuk-lekuk pada centre girder, side girder dan penumpu-penumpu.***



Penyebabnya adalah karena adanya deformasi buckling.

- ***Terjadinya penggelombangan pada pelat sekat.***



Penyebabnya adalah karena adanya deformasi buckling, deformasi memanjang.

- ***Terjadinya penggelombangan pada pelat kulit.***

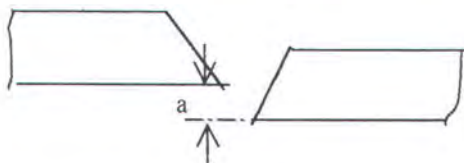


Penyebabnya adalah karena adanya deformasi buckling dan perubahan sudut pada penegar-penegarnya.

### III.3 Teknik Perbaikan Pada Penyimpangan Dimensi dan Bentuk.

Dalam melaksanakan perbaikan ada beberapa cara yang digunakan dalam mengatasi bentuk penyimpangan yang didasarkan atas jenis penyimpangan diantaranya :

#### ***Misalignment ( Ketidak lurusan sambungan Butt Join )***



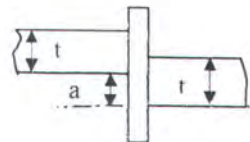
Ketidak lurusan pada bagian yang menerima beban utama

Perbaikan yang dapat dilakukan untuk batasan  $a > 0.15 t$  dan maksimum 3 mm adalah dengan merubah posisi kesalahan tersebut dan kemudian dilakukan pemasangan ulang.

Ketidak lurusan untuk bagian lain yang tidak menerima beban untuk  $a > 0.2 t$  dan maksimum 3 mm adalah dengan melakukan pemasangan ulang.



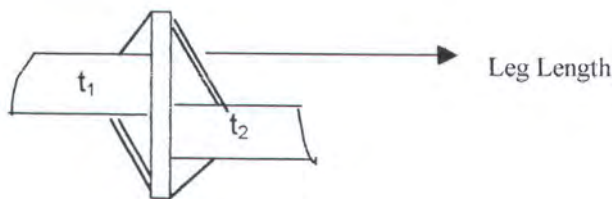
**Misalignmen ( ketidak lurusan pada sambungan Fillet )**



a : Perbedaan kelurusan

t : Tebal ( diambil yang paling tipis)

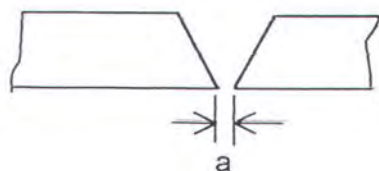
Perbaikan yang dapat dilakukan untuk batasan  $\frac{1}{3} t \leq a \leq \frac{1}{2} t$  adalah dengan cara memberikan penguat pada ujung-ujung sambungan seperti gambar dibawah ini :



Sedangkan untuk  $a > \frac{1}{2} t$  konstruksi tersebut harus dilakukan pemasangan ulang.

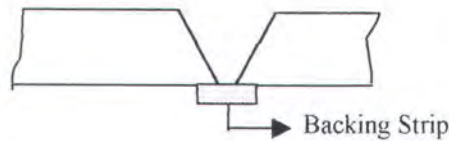
Untuk ketidaklurusan sambungan *fillet* pada bagian lain yang tidak menerima beban utama dengan batasan  $a > \frac{1}{2} t$  maka perbaikan yang dapat dilakukan dalah dengan memasang ulang konstruksi tersebut.

**Gap Atau Celah**



Untuk sambungan *Butt* pada Las tumpul *Manual welding* standar yang diijinkan adalah  $2 \leq a \leq 3,5$ .

Apabila  $5 < a < 16$  atau  $5 < a < t$ , maka perbaikan yang dapat dilakukan adalah dengan memberikan *Backing Strip*. Dan setelah dilas *backing strip* dibuang dan dilas setelah dilaksanakan *chipping*.

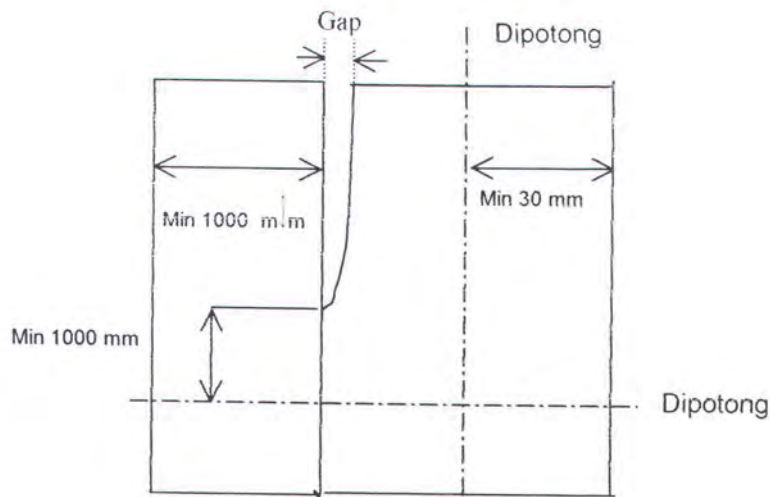


Apabila batasan terjadinya gap adalah  $16 < a < 25$  atau  $16 < a < t$ , maka harus dilakukan pengelasan built-up dengan persiapan *bevel*. Atau sebagian diganti dengan komponen baru.

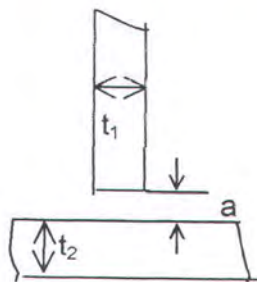
Apabila batasan  $a > 25$  atau  $A > t$  maka perbaikan yang dapat dilakukan adalah dengan mengganti sebagian pelat.

Pertimbangan yang harus dilakukan apabila dilakukan penggantian pelat adalah :

1. Berapa jarak lebar sambungan pelat minimum, minimum  $\pm 1000$  mm dari gap.
2. Apabila panjang gap pendek, maka panjang pelat pelat minimum  $\pm 1000$  mm q( minimum jarak antara buttt weld terhadap butt weld  $a \geq 30$  mm ).



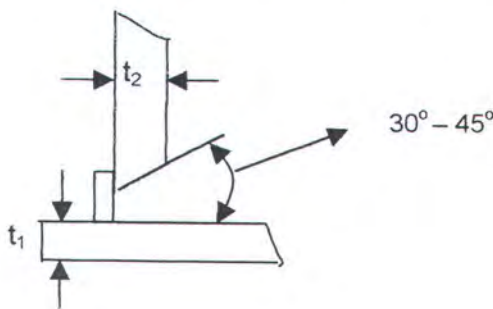
**Gap atau celah untuk sambungan fillet**



Standar yang diijinkan adalah  $a \leq 2$  dan batasan toleransi  $a \leq 3$  mm.

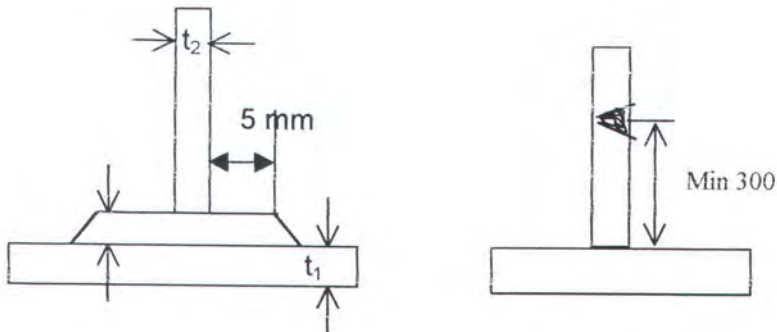
Cara perbaikan yang dapat dilakukan untuk penyimpangan  $3 < a < 5$ , maka kaki las disesuaikan dengan *rule* dan ditambah dengan  $a - 2$ .

Apabila penyimpangan  $5 < a \leq 16$  , maka perbaikan yang dapat dilakukan adalah dengan memberikan pengelasan dengan membuat bevel  $30^{\circ} \sim 45^{\circ}$ .





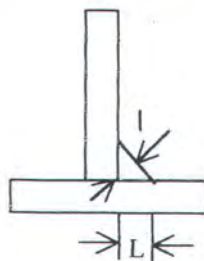
Dan apabila  $a > 16$  maka harus digunakan pelat sisipan atau sebagian ganti baru.



### **Leg Length**

*Leg length* ( Panjang kaki las ) dibandingkan dengan yang sebenarnya mempunyai batasan  $\geq 0,9 L$  dan  $0,9 \geq l$ .

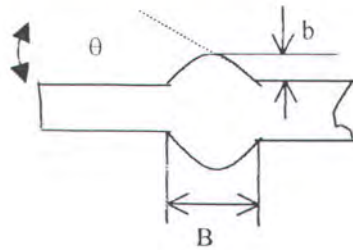
Cara perbaikan yang dapat dilakukan bila terjadi penyimpangan lebih besar dari batasan, bagian yang kurang harus ditambah las.



$L$  = Panjang kaki las.

$l$  = Ketinggian hasil las.

### ***Tinggi, Lebar, Dan Sudut Pengelasan***

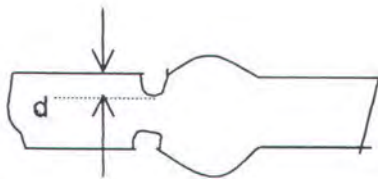


Batasan yang diijinkan  $\theta < 90^\circ$

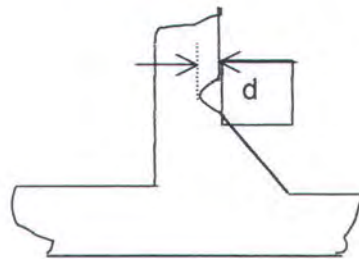
Cara perbaikan apabila  $\theta > 90^\circ$  adalah dengan menggerinda atau ditambah las sehingga sudutnya kurang dari  $90^\circ$ .

### ***Under Cut***

Penyimpangan dengan adanya takik las pada pelat kulit dan face pelat untuk



$$d \leq 0,5$$



$$d \leq 0,8$$

butt join mempunyai batasan  $< 90^\circ$  mm.

Cara perbaikan yang dapat dilakukan adalah diperbaiki dengan *elektrode* yang sesuai, hindari short baed untuk *higher tensile steel*.

## BAB IV

### STUDI KASUS PENYIMPANGAN DIMENSI DAN BENTUK PADA PROSES PRODUKSI KAPAL

#### IV.1 Persiapan Survei dan Pengambilan Data

Dari berbagai bentuk penyimpangan yang telah dipelajari dari literatur pada bab terdahulu, diperlukan studi lapangan guna memperoleh bentuk penyimpangan yang telah terjadi selama dilakukan proses produksi.

Dalam penulisan tugas akhir ini, survei dan pengambilan data dipilih galangan kapal PT. PAL INDONESIA. Hal ini didasarkan dengan kemampuan galangan dalam melakukan suatu proses produksi sangat didukung dengan peralatan dan kemampuan sumber daya manusia yang relatif cukup baik. Sehingga dalam melaksanakan setiap tahap dalam proses produksi kapal standar kerja dan standar peralatan dapat dipenuhi.

Bentuk survei yang dilakukan adalah dengan metode kuisiner (*terlampir*) yang diberikan kepada orang-orang dilingkungan *Quality Control (QC) / Quality Assurance (QA)* di PT. Pal Indonesia. Dan dari mereka akan diperoleh jawaban yang dibutuhkan dalam isian kuisiner tersebut.

Dalam penulisan kuisiner ini data-data awal yang digunakan adalah didasarkan literatur yang menjelaskan mengenai bentuk penyimpangan yang pernah terjadi selama proses produksi kapal. Sehingga dari data kuisiner tersebut akan didapat data lapangan mengenai bentuk penyimpangan untuk nantinya dapat diklasifikasikan ke dalam beberapa jenis penyimpangan.



Untuk mengklasifikasikan mengenai bentuk penyimpangan yang terjadi di kalangan penulis membagi berdasarkan tahapan proses yang dilakukan dalam produksi kapal digalangan, dimulai dari tahap *fabrikasi*, *sub-assembly*, *assembly* sampai dengan tahap *erection*. Sehingga berdasarkan tahap tersebut akan dapat diketahui pada tahap apa penyimpangan sering terjadi. Selanjutnya juga diklasifikasikan menurut komponen-komponen kapal, sehingga dapat diketahui pada komponen mana penyimpangan mudah terjadi, serta faktor –faktor yang sering menjadi penyebab terjadinya penyimpangan bentuk.

Disamping bentuk penyimpangan yang sering terjadi dalam proses produksi, dengan survei dilapangan ini diharapkan dapat diketahui pula bentuk dan jenis penyimpangan lain yang ada pada setiap tahap proses produksi kapal. Sehingga dari data-data dilapangan mengenai berbagai bentuk penyimpangan yang ada dapat kita standarisasikan menurut JSQS, yang kemudian dari penyimpangan tersebut dapat kita analisa untuk mendapatkan instruksi perbaikan yang akan dilakukan.

#### **IV.2 Hasil Survei Dan Interview Tentang Penyimpangan Dimensi Dan Bentuk**

Dari data survei dan pengambilan data di lapangan dalam hal ini PT. PAL INDONESIA, diperoleh berbagai bentuk penyimpangan yang terjadi selama proses produksi kapal. Selanjutnya data penyimpangan dimensi dan bentuk yang ada di lapangan diklasifikasikan berdasarkan teori bentuk penyimpangan produksi, bentuk penyimpangan yang terjadi tersebut dapat dikelompokkan menjadi beberapa jenis penyimpangan yaitu:

- Penyimpangan terhadap bentuk dan ukuran dari perencanaan.

- Penyimpangan terhadap bentuk dan ukuran karena deformasi.
- Penyimpangan terhadap bentuk dan ukuran karena penyusutan.

Setelah dapat diketahui berbagai bentuk penyimpangan yang terjadi, data-data tersebut diidentifikasi berdasarkan tahap proses produksi, mulai dari *Fabrikasi, sub-Assembly, Assembly, sampai dengan Erection*.

Setelah dapat diketahui tahap proses produksi, selanjutnya dikelompokkan berdasar seksi dan sub-seksi yang membagi antara proses pekerjaan dan komponen yang dikerjakan. Pembagian tersebut didasarkan atas standar JSQS

Data lengkap mengenai bentuk penyimpangan pada proses produksi kapal yang terjadi digalangan PT. PAL INDONESIA, dapat dilihat pada tabel lampiran.



## **BAB V**

### **PEMBUATAN MODEL PENYIMPANGAN DIMENSI DAN BENTUK**

#### **V.1. Hasil Analisa dari Standar JSQS dan Survei Lapangan**

Berdasarkan teori mengenai bagaimana proses produksi kapal dilaksanakan dan mempelajari hasil studi lapangan mengenai bentuk penyimpangan, maka dari berbagai bentuk penyimpangan tersebut dapat digeneralistik menjadi kasus umum dan sekaligus dicari perbaikan yang sesuai dengan standar yang dapat diterima.

Dengan diperolehnya data bentuk penyimpangan tersebut maka akan dapat dibuat kerangka pemikiran yang mampu mengakomodasikan hubungan antara permasalahan dan tindakan perbaikan. Adapun kerangka pemikiran tersebut adalah :

1. Berdasarkan teori mengenai bentuk penyimpangan pada proses produksi kapal untuk pemahaman permasalahan.
2. Studi lapangan untuk mengumpulkan data penyimpangan yang terjadi selama proses produksi kapal dilaksanakan.
3. Kesalahan – kesalahan yang timbul pada proses produksi kapal kemudian diidentifikasi dengan mengevaluasi besar penyimpangan antara dimensi desain dan actual, dan hasil pengukuran tersebut dimasukkan kedalam standar range dari JSQS atau memberi batas toleransi yang diperkenankan untuk masing-masing komponen yang diukur.
4. Bila konstruksi masih di dalam batas range, maka proses dilanjutkan, dan bila konstruksi diluar dari batas toleransi maka dilakukan pemberhentian



proses produksi dan dilaksanakan identifikasi penyimpangan dan kemudian akan didapat pemecahan untuk instruksi perbaikan yang dapat dilaksanakan.

Dengan memperhatikan kerangka berfikir seperti tersebut dapat diketahui secara jelas langkah-langkah yang diperlukan untuk mengidentifikasi penyimpangan dan perbaikan yang harus dilakukan.

Dari hasil survei lapangan dan identifikasi penyimpangan dengan menggunakan JSQS maka hasil analisa diperoleh hasil sebagai berikut:

### 1. Tahap Fabrikasi

- Bentuk Penyimpangan :

Terjadi kesalahan program pada *gas cutting* yang mengakibatkan material mengalami tekori / gap sebesar 8 mm.

Standar :

*JSQS, Dimension, size of member, general member compared with correct sizes. Standard range  $\pm 3,5$  mm. Tolerance limits  $\pm 5,0$  mm.*

Perbaikan :

Komponen di *built-up*.

- Bentuk Penyimpangan :

Terjadi kesalahan dimensi (tekori / terjadi gap) pada material pelat sebesar 15,40 dan 50 mm.

Standar :

*JSQS, Dimension, size of member, general member compared with correct sizes. Standard range  $\pm 3,5$  mm. Tolerance limits  $\pm 5,0$  mm.*

Perbaikan :

Komponen di built-up.

- Bentuk Penyimpangan :

Pada proses *nesting* terjadi kesalahan pada potongan 2 komponen dimana hasil potongan memotong kedua komponen sepanjang 40 mm. ( gambar dan program sudah benar ).

Standar :

*JSQS, Dimension, size of member, general member compared with correct sizes. Standard range  $\pm 3,5$  mm. Tolerance limits  $\pm 5,0$  mm.*

Perbaikan :

Pada komponen pertama pelat dipotong 260 mm kemudian disambung pelat baru sepanjang 300 mm, dan komponen kedua dibuatkan baru dengan data yang benar.

- Bentuk Penyimpangan :

Terjadi kesalahan penandaan pada posisi *IN* dari *design*. Tanda *bevel* terbalik.

Standar:

*JSQS, Dimension, length of taper  $\pm 0,5$  d. Tolerance limit  $\pm 1,0$  d, dimana d adalah tinggi bevel.*

Perbaikan :

Dibuatkan komponen baru dan dibevel sesuai dengan ukuran yang telah diperbaiki.

**2. Tahap Sub- Assembly dan Assembly**Bentuk Penyimpangan :

Terjadi tekor / gap antara *Tank Top* ke *Joint Block / Shell Joint* sebesar 15 - 30 mm.

Standar :

*JSQS, Fitting Accuracy, Gap before welding. Standard range  $a \leq 2$  , tolerance limits  $a \leq 3$ .*

Perbaikan :

Gap sebesar sampai 30 mm masih bisa ditutup oleh *margin* sebesar 50 mm. Dengan syarat ketinggian *deck* masih tetap terjaga.

- Bentuk Penyimpangan :

*Block RD plate 24* hasil tidak sesuai dengan gambar kerja. Ukuran yang benar 254 mm, sedangkan *lebar actual* 390 mm, jadi tidak cocok dengan *plate* dalamnya.

Standar :

*JSQS, Accuracy of dimensions. Breadth of Sub – assembly, Standard range  $\pm 4$  mm, Tolerance limits  $\pm 6$  mm.*

Perbaikan :



Komponen di built-up. Dengan dimensi yang benar yaitu 254 mm.

- Bentuk Penyimpangan :

Pada pelat *Wing Bhd* marking penempatan *Stifener* dari *nesting* tidak cocok dengan lubang *slot*, serta lebar *Wing Bhd* terjadi gap sebesar 40 mm.

Standar :

*JSQS, Size of member, Especially for the depth of floor and girder of double bottom compared with correct sizes. Standard range  $\pm 2,5$  mm. tolerance limits  $\pm 4,0$  mm.*

Perbaikan :

*Wing Bhd* dipotong 300 mm pada arah bawah kemudian disambung lagi sesuai kebutuhan.

- Bentuk Penyimpangan :

Terjadi gap antara sambungan *Wing Bhd* antara 13 - 43 mm.

Standar :

*JSQS, Fitting Accuracy, Gap before welding. Standard range  $a \leq 2$  , tolerance limits  $a \leq 3$ .*

Perbaikan :

Komponen *dibuild-up*. Dibuatkan *Wing Bhd* baru

- Bentuk Penyimpangan :

Pada block *Double Bottom* terjadi tekor / gap sebesar 20 mm terhadap *Bottom Shell*.

Standar :



*JSQS, Size of member, Especially for the depth of floor and girder of double bottom compared with correct sizes. Standard range  $\pm 2,5$  mm. tolerance limits  $\pm 4,0$  mm.*

Perbaikan :

Pada kiri dan kanan (P/S) dibuat sama yaitu satu *nesting*, apabila yang mengalami tekor/gap hanya bagian (S) saja kemungkinan ada kesalahan pada proses potong.

Dibuatkan material baru dengan merambu pada bagian (P) yang sudah benar.

- Bentuk Penyimpangan :

Pada *Block Double Bottom* sambungan antara dua plat *shell* terjadi gap sebesar 12 mm.

Standar :

*JSQS, Fitting Accuracy, Gap before welding. Standard range  $a \leq 2$  , tolerance limits  $a \leq 3$ .*

Perbaikan :

Salah satu dari pelat diadakan *built-up*. Dengan memotong salah satu pelat dengan jarak minimum 300 mm dan kemudian dipasang pelat baru.

- Bentuk Penyimpangan :

Pada komponen *Wing Bhd* potongan NC Cutting antara tiga jarak *frame* berbelok keatas sehingga terjadi gap sebesar 10 mm.

Standar :

*JSQS, Fitting Accuracy, Gap before welding. Standard range  $a \leq 2$ , tolerance limits  $a \leq 3$ .*

Perbaikan :

Proses pengelasan di *built-up*.

- Bentuk Penyimpangan :

Pada pelat *Deck* posisi *Aft* lebar 2750, keadaan aktual 2650, jadi terjadi gap sebesar 100 mm.

Standar :

*JSQS, Fitting accuracy, gap before welding. Butt weld, Standard range  $2 \leq a < 3,5$ , Tolerance limits  $a \leq 5$ .*

Perbaikan :

Untuk  $a > 25$ , pelat harus dipotong sebagian dengan jarak minimum 300 mm dan diganti pelat baru.

- Bentuk Penyimpangan :

Pada salah satu *frame (P/S)*, terjadi gap sebesar 30 mm terhadap *Tank Top*.

Standar :

*JSQS, Fitting Accuracy, Differences between the beam and the frame. Standard range  $a \leq 3$ , tolerance limits  $a \leq 5$ .*

Perbaikan :

Komponen dipotong sebagian dengan jarak minimum 300 mm dan diganti pelat baru.

- Bentuk Penyimpangan :



Lebar komponen gambar 1000 mm, sedangkan material komponen lebar 950 mm. Jadi terjadi tekor sebesar 50 mm.

Standar :

JSQS, *Accuracy of dimensions. Breadth of Sub – assembly, Standard range  $\pm 4$  mm, Tolerance limits  $\pm 6$  mm.*

Perbaikan :

Dibuatkan material baru sesuai *marking list*.

- Bentuk Penyimpangan :

Terjadi tekor/gap sebesar 50 mm terhadap *Bottom Shell (Bilga)*.

Standar :

JSQS, *Fitting Accuracy, Gap before welding. Standard range  $a \leq 2$  , tolerance limits  $a \leq 3$ .*

Perbaikan :

Terjadi kesalahan pada proses potong sehingga perlu dibuatkan material baru.

### 3. Tahap Erection

- Bentuk Penyimpangan :

Terjadi kesalahan pemotongan tirus (*Okobei*) pada sambungan dua pelat.

Standar :

JSQS, *Dimension, Length of taper. Standard range  $\pm 0,5 d$ , tolerance limits  $\pm 1,0 d$ . Dimana  $d$  adalah tinggi tinggi bevel / tirus.*

Perbaikan :

Apabila tidak ada kesalahan dimensi, pelat yang belum dibevel / tirus pelat dapat dibevel sesuai dengan ukuran yang telah ditentukan.

- Bentuk Penyimpangan :

Terjadi kesalahan pada sambungan dua pelat dimana pelat pertama dengan tebal 21 mm dan pelat kedua dengan tebal 12 mm belum dipotong tirus (*Okobei*).

Standar :

*JSQS, Dimension, Length of taper. Standard range  $\pm 0,5 d$ , tolerance limits  $\pm 1,0 d$ .* Dimana d adalah tinggi tinggi bevel / tirus.

Perbaikan :

Pada kedua pelat dapat dibuatkan *bevel* / tirus, dan kedua pelat dapat disambung.

- Bentuk Penyimpangan :

Terjadi kesalahan dimensi pada *Bracket*.

Standar :

*JSQS, Dimension, size of member, general member compared with correct sizes. Standard range  $\pm 3,5 mm$ . Tolerance limits  $\pm 5,0 mm$ .*

Perbaikan :

Dilakukan pemotongan 300 mm dari bawah kemudian disambung sesuai kebutuhan.

- Bentuk Penyimpangan :

Terjadi kekurangan dimensi pada *Stiffener* sebesar 35 mm (P) dan 55 mm (S).

Standar :

*JSQS, Dimension, size of member, general member compared with correct sizes. Standard range  $\pm 3,5$  mm. Tolerance limits  $\pm 5,0$  mm.*

Perbaikan :

Dilakukan pemotongan sebesar 300 mm kemudian disambung dengan *Stiffener* yang sama sesuai kebutuhan. ( $\pm 300$  mm).

- Bentuk Penyimpangan :

Tinggi sambungan pelat dari *wash Deck* kurang sebesar 150 mm dari yang direncanakan pada *Drawing*.

Standar :

*JSQS, Fitting Accuracy, Gap before welding. Standard range  $a \leq 2$  , tolerance limits  $a \leq 3$ .*

Perbaikan :

Pengerutan pelat kemungkinan pada proses pemanasan dan pengelasan.

*Mould Loft* agar memberikan margin yang cukup, terutama untuk bagian yang kelengkungannya besar.

- Bentuk Penyimpangan :

Terjadi ketidak sesuaian antara *Wing Bulkhead* pada *Transverse*, besar penyimpangan sebesar 30 - 45 mm. Kesalahan pada lebar *joint block* bagian bawah (*drawing* sebesar 1600 mm), yaitu kurang dari 1600.



Standar :

JSQS, *Accuracy of dimensions. Breadth of Sub – assembly, Standard range  $\pm 4$  mm, Tolerance limits  $\pm 6$  mm.*

Perbaikan :

Perbedaan tersebut terjadi karena ada beda tebal pelat *inner Wall* dan adanya kemiringan. Tindakan yang dilakukan adalah dengan mengganti bagian bawah sisi dalam *transverse*.

- Bentuk Penyimpangan :

Terjadi Gap antara sambungan dua Plat *Shell Block* antara 8 - 20 mm.

Standar :

JSQS, *Fitting Accuracy, Gap before welding. Standard range  $a \leq 2$  , tolerance limits  $a \leq 3$ .*

Perbaikan :

Setelah dilakukan pengecekan pada komputer ternyata untuk kanan (S) dan kiri (P) sama. Kemungkinan terjadi kesalahan pada proses sebelumnya.

- Bentuk Penyimpangan :

Plat *Shell Block* DB terjadi gap sebesar antara 0 - 15 mm dan Plat *Shell Block* lain antara 0- 10 mm.

Standar :

JSQS, *Fitting Accuracy, Gap before welding. Standard range  $a \leq 2$  , tolerance limits  $a \leq 3$ .*

Perbaikan :

Setelah dilakukan pengecekan pada komputer ternyata untuk kanan (S) dan kiri (P) sama. Kemungkinan terjadi kesalahan pada proses sebelumnya.

- Bentuk Penyimpangan :

Terjadi lekuk pada plat kulit yang dalam

Standar :

*JSQS : Frame & Long for deviation from correct form - standard range 3.0 mm, tolerance limits 5.0 mm.*

Perbaikan :

Plat yang lekuk diluruskan kemudian di *fairing*.

- Bentuk Penyimpangan :

Pada Block terjadi gap sebesar 3, 5 mm ke arah kanan (S).

Standar :

*JSQS, minimum distance of weld to adjacent weld for butt weld; tolerance limits 30 mm.*

Perbaikan :

Plat dipotong sepanjang gap yang tidak memenuhi toleransi kemudian dipasang plat ukuran 350 x 6 jarak *frame*.

- Bentuk Penyimpangan :

*Hull Construction* pada *Engine Girder* terjadi gap pada sambungan tumpul antar *floor* dan *engine girder*.

Standar:

*JSQS. Fitting accuracy for gap before welding standard range  $a < 2$  , tolerance limits  $a < 3$  .*

Perbaikan :

*Floor* dipotong dengan jarak minimum 300 mm kemudian dipasang plat baru.

- Bentuk Penyimpangan :

Terjadi gap sambungan tumpul pada sambungan *Bracket* melebihi 10 mm.

Standar:

*JSQS. Fitting accuracy for gap before welding for butt weld - standard range  $2 < a < 3,5$  , tolerance limits  $a < 5$  .*

Perbaikan :

*Bracket* dipotong sepanjang 300 mm kemudian dipasang *bracket* baru.

- Bentuk Penyimpangan :

Terjadi Deformasi pada plat tepi yang terletak pada sisi kanan (S).

Standar:

*JSQS. Frame and Long, Deviation from correct form - standard range 3 m, tolerance limits 5 mm.*

Perbaikan :

Plat diluruskan dan urutan pengelasan dilakukan dengan benar.



- Bentuk Penyimpangan :

Pada sekat melintang sambungan antar plat tersebut mengalami *Misalignment* sepanjang  $\pm 350$  mm dan terjadi gap pada *deck* sepanjang  $+ 300$  mm dan tinggi 20 mm.

Standar:

*JSQS. Fitting accuracy, fillet weld - standard range  $a < 2$  , tolerance limits  $a < 3$ .*

Perbaikan :

Konstruksi di *built-up* dan diadakan pelurusan kemudian digerinda.

- Bentuk Penyimpangan :

Terjadi *misalignment* untuk prose *joint erection* pada *upper deck*.

Standar:

*JSQS. Fitting accuracy, fillet weld - standard range  $a < 2$  , tolerance limits  $a < 3$ .*

Perbaikan :

Pada *side shell* yang lari keluar *upper deck* dipotong dan pada *side shell* yang masuk kedalam ditarik keluar.

- Bentuk Penyimpangan :

Terjadi gap pada sambungan T antara *floor* dan *side girder double bottom*.

Standar:

*JSQS. Fitting accuracy for gap before welding - standard range  $a < 2$  , tolerance limits  $a < 3$  .*

Perbaikan :

*Floor* dipotong sepanjang 300 mm dan dipasang dengan plat baru.

## V.2. Model Alat Bantu untuk Standarisasi Penyimpangan dan Instruksi Perbaikan

Berdasarkan teori mengenai bentuk penyimpangan dan setelah mempelajari hasil studi lapangan di PT. PAL INDONESIA, dan telah dijelaskan pada bab terdahulu. Pada bab ini akan dijelaskan bagaimana proses standarisasi penyimpangan dan perbaikan.

Proses analisa ini diarahkan pada identifikasi bentuk penyimpangan yang dapat digeneralistik menjadi bentuk umum sehingga dapat dicari penyelesaian masalah berupa instruksi perbaikan yang dapat diterima secara teknis.

Dengan memperhatikan pembahasan sebelumnya, maka disusun suatu model yang mampu mengakomodasikan hubungan antara bentuk penyimpangan, *standar range*, dan *toleransi* yang diijinkan. Sehingga dapat diidentifikasi jenis penyimpangan yang terjadi yang nantinya akan menghasilkan instruksi perbaikan yang harus dilakukan.

Pemodelan ini berupa *database* mengenai berbagai bentuk penyimpangan yang telah terjadi dan perbaikan yang telah dilakukan selama proses produksi kapal. *Database* ini diperoleh dari studi literatur dan studi lapangan dimana dari data tersebut di-identifikasi dengan standar JSQS.

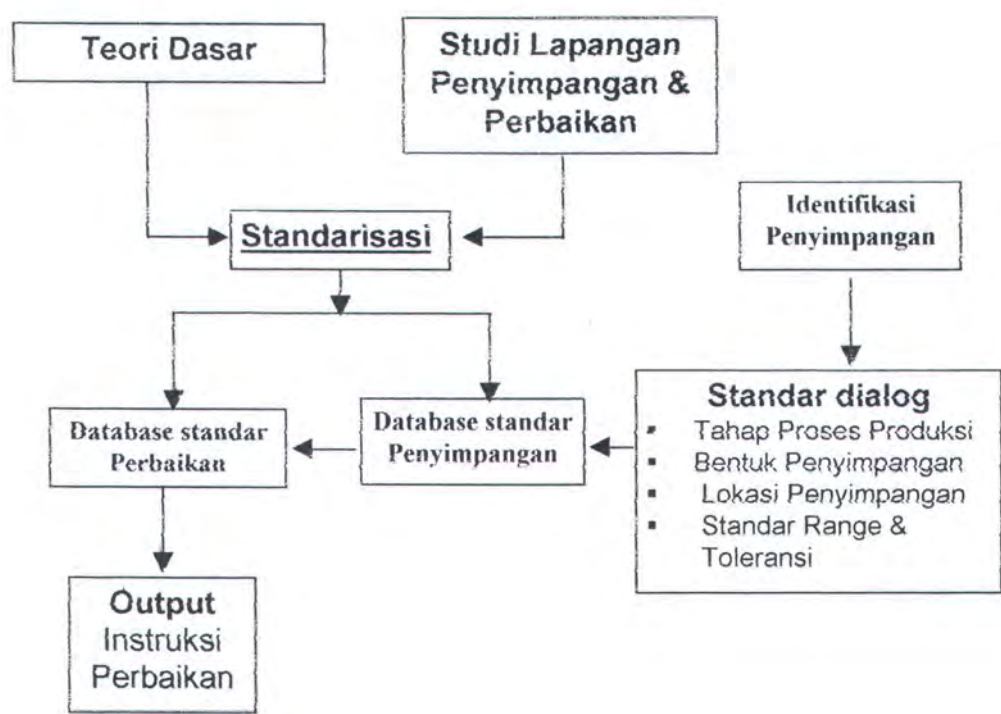
Selanjutnya apabila terjadi penyimpangan, pemakai dapat mengidentifikasikan bentuk penyimpangan berdasar standar dialog berupa :

- Tahap pemeriksaan
- Lokasi penyimpangan
- Jenis penyimpangan
- Standar yang diijinkan



Dari identifikasi tersebut maka kita dapat menentukan perbaikan yang dapt dilakukan.

Model alat bantu untuk standarisasi penyimpangan dan instruksi perbaikan dapat dibuat alur sederhana sebagai berikut :



Dengan didasarkan dari model tersebut maka dibuatknya program komputer akan sangat membantu dalam proses pelaksanaan perbaikan penyimpangan.

Dengan adanya standarisasi penyimpangan dan perbaikan pada proses produksi kapal ini akan mempunyai keuntungan dibanding dengan proses manual yang selama ini masih digunakan di banyak galangan di Indonesia Perbedaan dari kedua proses tersebut antara lain:



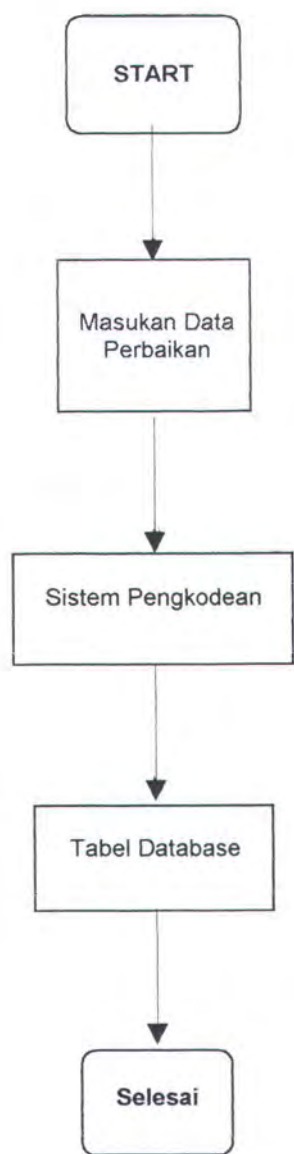
Manual	Komputerisasi Standar Penyimpangan dan Perbaikan
<b>Waktu:</b>  Relatif lama	  Relatif lebih Cepat
<b>Operator:</b>  Tergantung keahlian Operator	  Tidak bergantung keahlian operator
<b>Keakuratan:</b>  Kurang akurat karena tergantung operator	  Lebih akurat karena tidak bergantung operator
<b>Mutu:</b>  Sering berubah untuk tiap penyimpangan yang sama	  Mutu lebih terjaga.

Dengan demikian adanya standarisasi penyimpangan dan perbaikan sangat menunjang produktifitas suatu galangan.

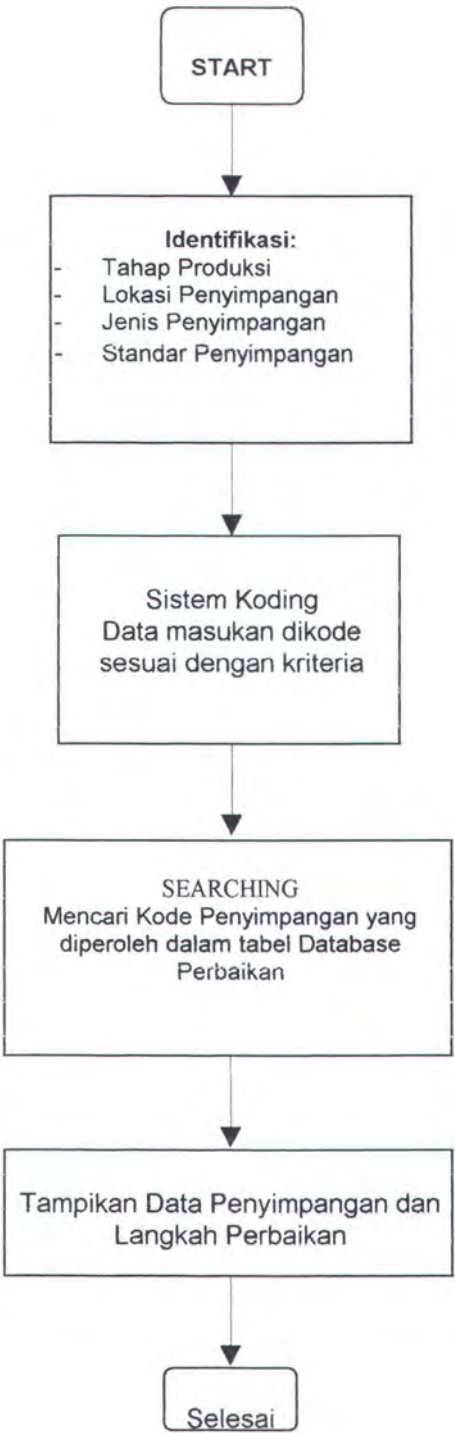
V.3. Flowchart Program Komputer untuk Identifikasi Penyimpangan dan Instruksi Perbaikan

Pembuatan Program Komputer dibagi atas Pembuatan Program *Database* dan Program Identifikasi Penyimpangan

Flowchart Program *Database*



Flowchart Identifikasi Penyimpangan





## BAB VI

# PEMBUATAN PROTOTYPE PROGRAM PENYIMPANGAN BENTUK DAN INSTRUKSI PERBAIKAN

### VI.1. ALASAN PEMILIHAN BAHASA PEMROGRAMAN

Dalam pemilihan bahasa pemrograman kita dihadapkan pada berbagai *software* bahasa pemrograman komputer yang ada. Saat ini terdapat berbagai macam *software* komputer yang dapat kita bagi menjadi 4 jenis berdasarkan kegunaannya yaitu :

#### 1. **OPERATING SYSTEM (Sistem Operasi) :**

*Operating System* merupakan dasar dari suatu perangkat lunak untuk memulai menjalankan komputer. Tanpa *operating system* maka komputer tidak dapat digunakan. Contoh *operating system* diantaranya:

- *MS-DOS (Microsoft Disk Operating Sistem)*
- *OS/2 (Operating Sistem 2)*
- *UNIX*
- *XENIX*
- *MS-WINDOWS*

#### 2. **UTILITY (Software Pembantu)**

*Utility* merupakan suatu elemen *software* yang bertugas mengerjakan suatu pekerjaan *minor* dalam pengoperasian *operating system* seperti menghapus *file*, *mengcopy file*, memasang *password* dan lain sebagainya.

Contoh dari *utility* ini diantaranya adalah:

- *PC TOOLS*
- *Norton Utility*
- *Norton Comander*
- *Advance Diagnostic*

### 3. **PROGRAM PACKAGE (Paket Program)**

*Program Package* merupakan suatu paket program yang sudah jadi dan siap dioperasikan. Namun kadangkala terdapat suatu kemampuan lain dari *program package* tersebut seperti adanya kemampuan untuk membuat program aplikasi yang lebih mudah digunakan daripada bila menggunakan program package secara langsung. Contoh dari *program package* yaitu:

- Untuk pengolah kata (*Word Processor*), misalnya:
  - a. *Wordstar*
  - b. *Microsoft Word*
  - c. *Amipro*
  - d. *ChiWriter*
- Untuk pengolah tabel perhitungan (*Sreadsheet*), misalnya:
  - a. *Lotus 123*
  - b. *Quattro Pro*
  - c. *Microsoft Excel*
  - d. *Simpony*



- Untuk pembuatan artikel (*Desktop Publisher*), misalnya:
  - a. *PageMaker*
  - b. *Ventura*
  - c. *News Room*
  
- Untuk Presentasi, misalnya:
  - a. *Microsoft Power Point*
  - b. *Harvard Graphic*
  
- Untuk Pengolah *Database (Database Processor)*, misalnya:
  - a. *DBase III Plus*
  - b. *Foxbase*
  - c. *Clipper*
  - d. *FoxPro*
  - e. *Acces*
  
- Untuk \*pembuatan gambar teknik / *CAD (Computer Aided Design)*, misalnya :
  - a. *Autocad*
  - b. *ProDesign*
  - c. *Dimensi Architectural Home*
  
- Untuk perhitungan statistik, misalnya :
  - a. *SAS*



- b. *TORA*
- c. *SPSS*
- d. *StatPek*
- e. *Minitab*

- Untuk perhitungan akuntansi, misalnya :
  - a. *DEA ( Dac Easy Accounting )*
  - b. *Quicken*

#### **4. PROGRAM LANGUAGE ( Bahasa Pemrograman ) :**

*Program Language* merupakan *software* yang digunakan untuk pembuatan program-program baik dalam bahasa tingkat tinggi maupun tingkat rendah. Bahasa tingkat tinggi mempunyai perintah yang mirip dengan bahasa manusia, mudah dimengerti namun kemampuannya terbatas. Sedang bahasa tingkat rendah mempunyai perintah yang berbeda dengan bahasa manusia, lebih sulit dimengerti, namun kemampuannya tinggi. Contoh bahasa tingkat tinggi adalah :

- a. *BASIC ( Beginner All Purpose Symbolic Code )*
- b. *PASCAL*
- c. *COBOL (Common Bussiness Oriented Language )*
- d. *Borland Delphi*

Sedang bahasa tingkat rendah adalah :

- a. *Assembler*
- b. *C++*

Karena itu setelah melalui serangkaian evaluasi maka dipilih Program

Language *BORLAND DELPHI for WINDOWS RELEASE 3.0* untuk membuat program komputer Standarisasi penyimpangan dan perbaikan pada proses produksi kapal karena *BORLAND DELPHI for WINDOWS 95 RELEASE 3.0*

Mempunyai kemampuan sebagai berikut :

1. Merupakan bahasa pemrograman yang mempunyai kemampuan pada pengolahan *database*, baik di dalam menata maupun mengelola informasi *database* yang kini tersimpan dalam bentuk daftar, kartu, map, dan tabel. Sehingga pekerjaan *managemen database* dapat disederhanakan.
2. Waktu yang dibutuhkan untuk pengaksesan program kira-kira 8 (delapan) kali lebih cepat.
3. Mempunyai fasilitas untuk *mouse* yang dapat mempercepat pemilihan menu.
4. Mempunyai fasilitas 140 instruksi lebih banyak.
5. Mempunyai fasilitas *aplikasi visual* yang menarik yang tercakup dalam *Screen Builder, Menu Builder, Report Writer*.
6. Mempunyai fasilitas dasar dari Sistem Operasi *Microsoft Windows* yaitu DDE (*Dinamic Data Exchange*) yang menyebabkan *Microsoft Borland Delphi for Windows* ini dapat berkomunikasi dengan program aplikasi *Windows* yang lain sehingga dapat menukar data dengan *spereads sheet, word processor, sistem e-mail* dan lain-lain, serta *OLE (Object Linking and Embedding)* yang digunakan untuk berhubungan dengan program aplikasi *Windows* yang lain kemudian mengambil tampilan program aplikasi tersebut seperti tabel, bunyi, *spread sheet*, gambar dan lain-lain.
7. Mempunyai kemampuan membuat program pengolah *database* melalui cara yaitu:
  - Melalui penulisan *Syntak* dalam jendela *Command*



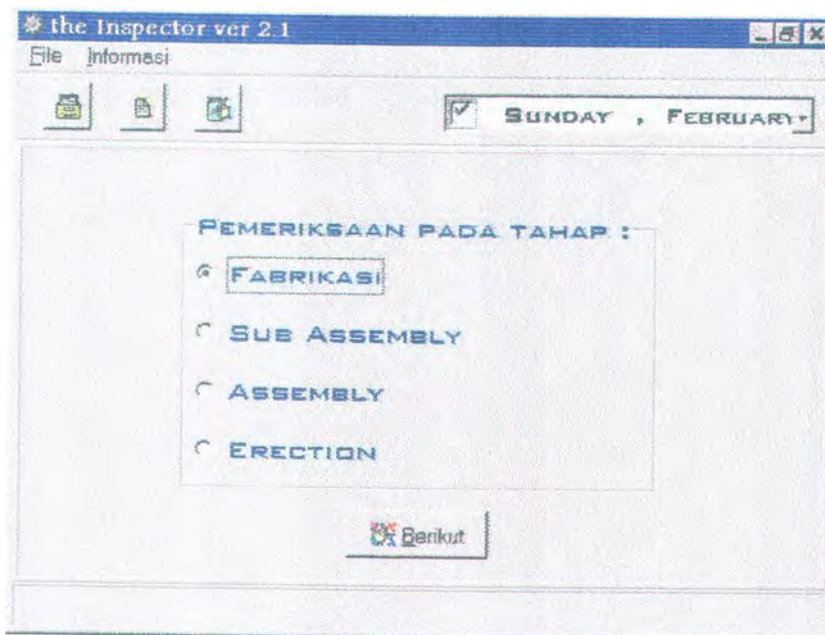
- Melalui pemakaian *menu Screen Builder, Menu Builder, dan Report Writer.*
9. Mempunyai kemampuan menyediakan jenis, ukuran *font*, serta *grafis* yang relatif lengkap.

## VI.2. TAHAPAN KONSULTASI

Dalam menjalankan prototype program ini digunakan program *Borland Delphi for Windows 95 Release 3.0.*

Langkah-langkah untuk menjalankan *Prototype Program* Komputerisasi Standar Penyimpangan dan Perbaikan Pada Proses Produksi Kapal adalah sebagai berikut ;

1. Pilih Program *The Inspector* pada *Windows Explorer*. Program akan menampilkan tampilan utama seperti gambar 6.1



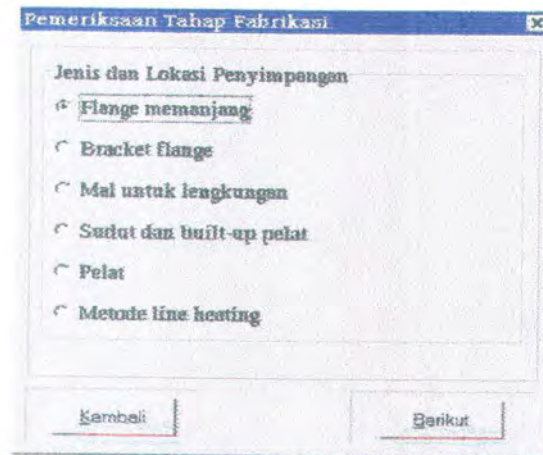
Gambar 6.1 Tampilan Utama Prototype Program



Pada tahap ini pemakai menentukan tahap proses produksi. Pada operasi ini pilihan akan terbagi menjadi 4 pilihan yaitu:

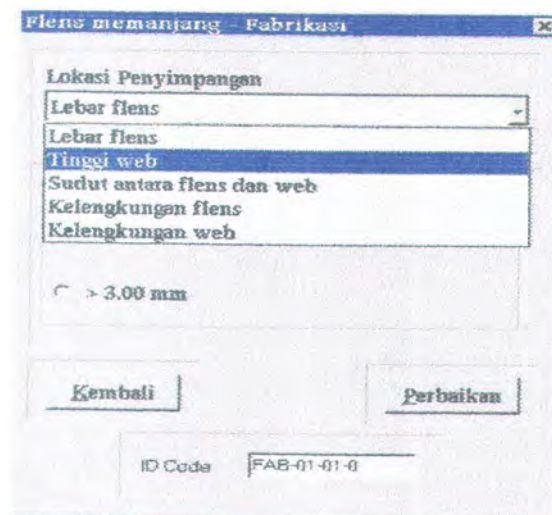
### **FABRIKASI**

2. Setelah ditentukan tahap ini, tahap konsultasi selanjutnya adalah menentukan jenis dan lokasi penyimpangan. Program akan menampilkan seperti pada gambar 6.2 dibawah ini



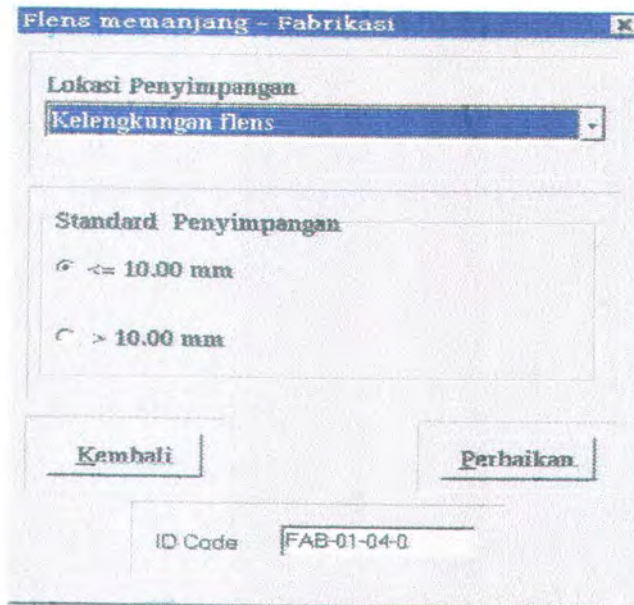
Gambar 6.2 Pemeriksaan Tahap Fabrikasi

3. Pilih salah satu dari Jenis dan Lokasi Penyimpangan. Program akan menampilkan seperti pada gambar 6.3 dibawah ini:



Gambar 6.3 Penyimpangan pada Flens Memanjang.

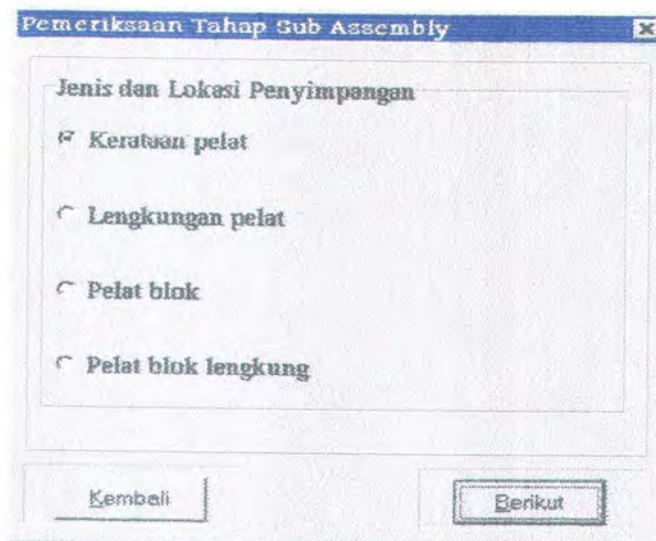
4. Pilih Salah satu dari lokasi Penyimpangan. Tampilan program akan memberikan pilihan Standar Penyimpangan yang diijinkan, seperti tampak pada gambar 6.4 dibawah ini :



Gambar 6.4 Lokasi dan Standar Yang Diijinkan

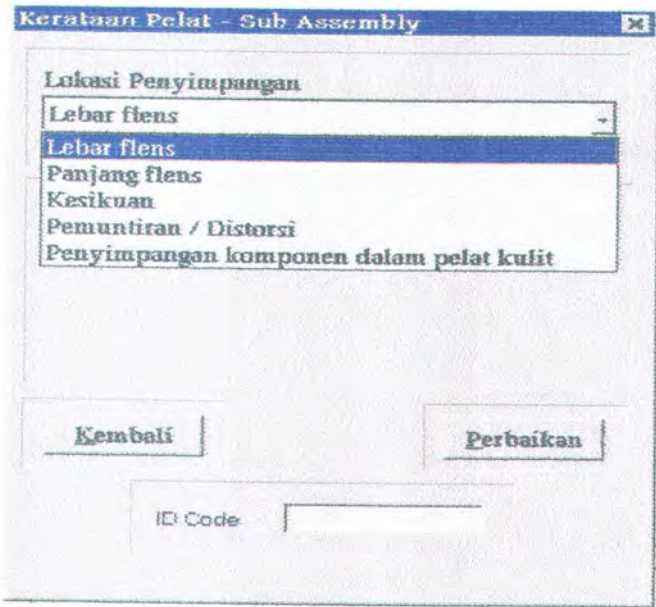
### **SUB - ASSEMBLY**

1. Pilih *Sub-Assembly* pada menu tahap Pemeriksaan. Tampilan Program seperti tampak pada gambar 6.5



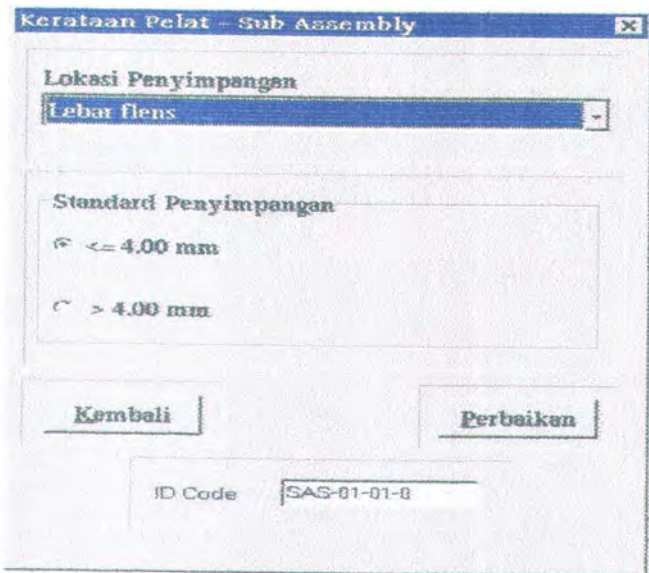
Gambar 6.5 Pemeriksaan Tahap Sub Assembly

- 2. Pilih salah satu jenis dan lokasi Penyimpangan. Tampilan Program akan tampak seperti gambar 6.6



Gambar 6.6 Lokasi dan Standar yang diijinkan.

- 3. Pilih salah satu lokasi penyimpangan untuk menentukan standar penyimpangan. Tampilan program tampak seperti gambar 6.7 seperti dibawah ini:

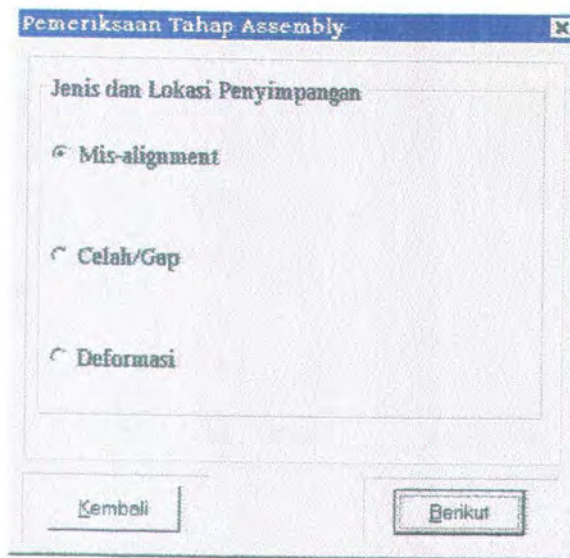


Gambar 6.7 Lokasi dan Standar Penyimpangan yang diijinkan



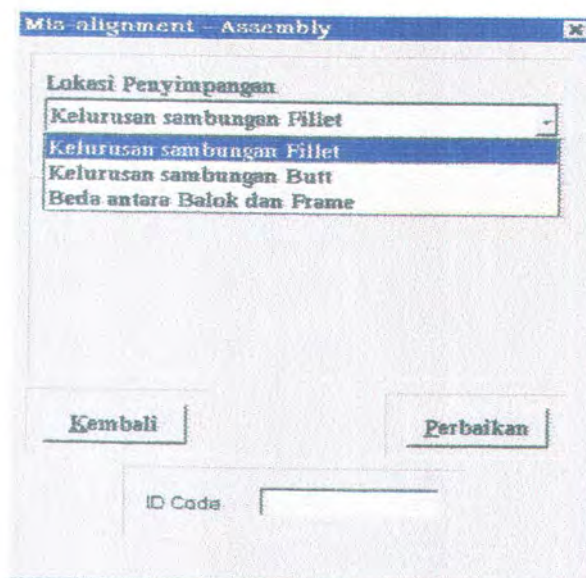
## ASSEMBLY

1. Setelah ditentukan tahap pemeriksaan, pilih Jenis dan Lokasi Penyimpangan . Tampilan program tampak seperti gambar 6.8 dibawah ini:



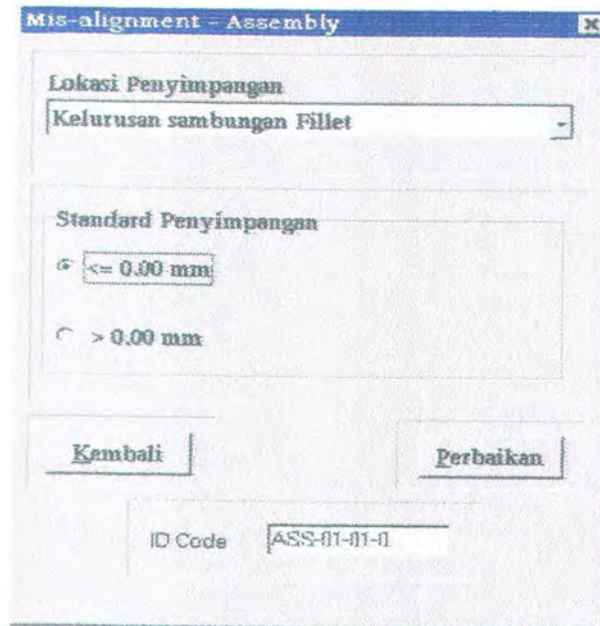
Gambar 6.8 Jenis dan Lokasi Penyimpangan

2. Pilih Lokasi Penyimpangan. Tampilan program tampak seperti gambar 6.9 dibawah ini:



Gambar 6.9 Lokasi Penyimpangan.

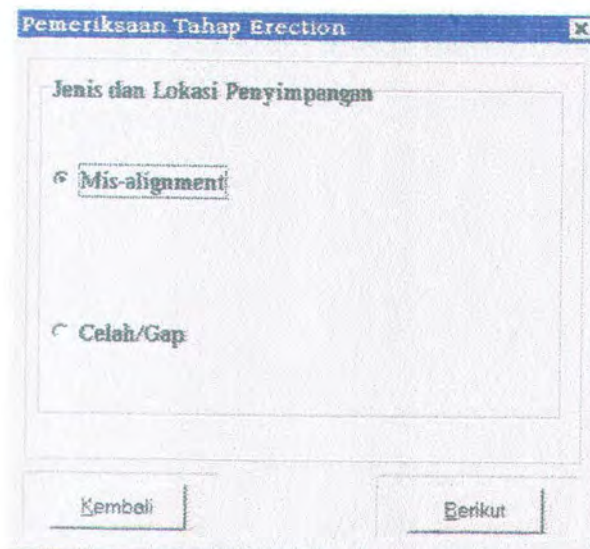
3. Pilih Standar yang diijinkan. Tampilan program tampak seperti gambar 6.10 dibawah ini :



Gambar 6.10. Standar yang diijinkan.

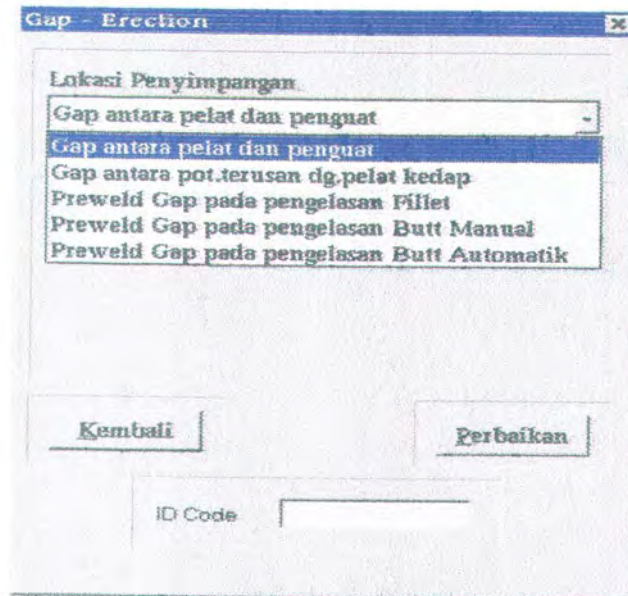
## ERECTION

1. Pilih tahap Erection pada pilihan tahap Pemeriksaan. Tentukan Jenis dan Lokasi Penyimpangan. Tampilan Program tampak seperti gambar 6.11 dibawah ini:



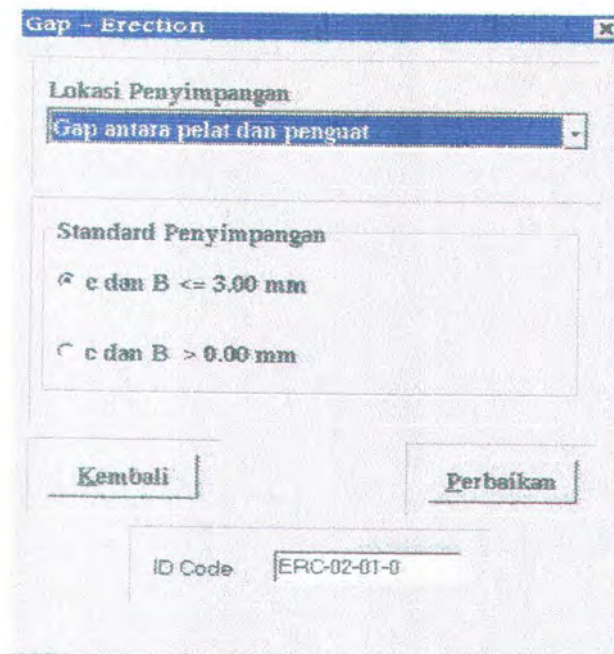
Gambar 6.11 Lokasi dan Jenis Penyimpangan

2. Pilih lokasi penyimpangan Program tampak seperti gambar 6.12 dibawah ini:



Gambar 6.12 Lokasi Penyimpangan.

3. selanjutnya tentukan standar yang diijinkan. Tampilan program akan tampak seperti gambar 6.13 dibawah ini :



Gambar 6.13 Standar Penyimpangan yang diijinkan



### VI.3. HASIL KONSULTASI

Setelah dilakukan identifikasi dan diperoleh instruksi perbaikan yang harus dilakukan maka hasil dari instruksi perbaikan tersebut dapat dicetak kedalam form prosedur perbaikan, seperti tampak pada gambar 6.14 dibawah ini:

INSPEKTOR V2.4		Surat, 05 Februari 1995
PROSEDUR PERBAIKAN		
Kode Taklang Perbaikan	P/B	
Taklang Perbaikan	Perbaikan	
Kode Indikasi	di	
Lokasi Deforansi	Pangkal runway	
Kode Jenis Deforansi	di	
Jenis Deforansi	Lubang runway	
Deforansi Standard	1 - 4 mm	
Kelas deforansi	Deforansi standar	
Deforansi yang terjadi	Deforansi Perbaikan	
1. Masalah ini	1. Segan yang berakibat di bagian runway di bagian yang di deforansi	

Gambar 6.14 Form Instruksi Perbaikan

## **BAB VII**

### **DISKUSI DAN REKOMENDASI**

Dalam tugas akhir ini, kami melaksanakan pengukuran dan pendataan identifikasi permasalahan khususnya pada penerapan analisa perbaikan. Selanjutnya hasil-hasil pengukuran dan informasi yang diperoleh tersebut diidentifikasi dan diklasifikasikan menurut sistem evaluasi dan analisa pada standar JSQS.

Pada setiap tahap proses produksi kapal diperlukan analisa mengenai data variasi ukuran penyimpangan dimensi dan bentuk, dengan didasarkan standar JSQS. Dengan dapat diketahui besarnya ukuran penyimpangan dimensi dan bentuk yang terjadi , dapat ditentukan bagaimana perbaikan-perbaikan yang harus dilakukan guna mencapai tingkat ketepatan ukuran yang ditentukan.

Dengan adanya standarisasi bentuk penyimpangan dan perbaikan, produktifitas galangan akan semakin baik dan waktu pembangunan kapal akan lebih singkat.

Standarisasi ini selain akan mempercepat dalam penentuan perbaikan yang harus dilakukan, juga akan memberi jaminan akan mutu dari perbaikan yang dilakukan dan menghindari ketergantungan proses produksi terhadap kemampuan perseorangan dalam menentukan instruksi perbaikan.

Dalam tugas akhir ini hanya ditinjau bentuk penyimpangan yang terjadi pada konstruksi lambung kapal. Sedang pada identifikasi masalah tidak memasukkan pendataan kegiatan produksi yang dipengaruhi oleh peralatan,

tenaga kerja dan spesifikasi material secara terperinci, sehingga dalam pengambilan keputusan untuk perbaikan tidak diidentifikasi mengenai penyebab dari terjadinya penyimpangan. Dengan demikian dapat ditindaklanjuti guna lebih meningkatkan hasil instruksi perbaikan yang lebih baik.



## **BAB VIII**

### **KESIMPULAN DAN SARAN**

Setelah mempelajari dan menganalisa uraian diatas tentang komputerisasi standar penyimpangan dan perbaikan pada proses produksi kapal, maka dapat disimpulkan bahwa :

1. Pada setiap tahap proses produksi kapal sering terjadi penyimpangan, mengingat pada setiap tahap produksi komponen yang terlibat dalam pembangunan sangat banyak dan kompleks. Sehingga memperbesar peluang terjadinya penyimpangan.
2. Dalam setiap pelaksanaan pembangunan kapal perlu dilakukan pemeriksaan berkala dan kontinyu dan pelaksanaan pemeriksaan itu harus dilakukan oleh pihak yang berkait yaitu:
  - Owner Surveyor
  - Biro Klasifikasi
  - Galangan

Sehingga dengan adanya standarisasi maka ketiga pihak tersebut menggunakan acuan yang sama sebagai pedoman standar.

3. Dengan standarisasi penyimpangan dan perbaikan akan mempermudah dan mempercepat dilakukannya perbaikan akibat adanya penyimpangan selama dilakukan proses produksi.
4. Pada setiap tahap produksi kapal digalangan ternyata ditemukan beberapa penyimpangan yang secara umum dapat generalisir.

Saran-saran yang dapat kami sampaikan adalah sebagai berikut :

1. Mengingat masih seringnya terjadi penyimpangan selama proses produksi kapal digalangan kapal, maka perlu adanya standarisasi penyimpangan dan perbaikan, untuk menentukan perbaikan yang harus dilakukan bila terjadi penyimpangan selama proses produksi kapal.
2. Untuk meningkatkan kemampuan, kecepatan dan hasil kerja standarisasi bentuk penyimpangan dan perbaikan pada proses produksi kapal, usaha komputerisasi yang berpijak pada pengolahan *database* dapat dilakukan dengan pembuatan program komputer.

## DAFTAR PUSTAKA

"**Japan Shipbuilding Quality Standart**", Quality Assurance Manual, PT.PAL Indonesia , 1980.

Andrew Wozniewicz with Namir Shammass, **Teach Your Self Delphi in 21 Days**, SAMS Publishing 1995.

Antony Pranata, **Pemrograman Borland Delphi**, Penerbit Andi Jogjakarta.

Antony Pranata, **Tip dan Trik Pemrograman Delphi**, Penerbit Adni Jogjakarta.

Fakultas Teknologi Kelautan, Institut Teknologi Sepuluh Nopember, **Laporan Kegiatan Pengawasan Pembangunan Kapal** Penelitian Kelautan LIPI Tahap I di PT PAL Indonesia (Persero)

Mac Kenzie Wilson, Bsc, Msc, Ceng, MRINA, GradIMechE and L.R. Tithecott: paper Appledore Shipbuilder - **A New Concept in Shipbuilding?**

Modul Pelatihan **Sistim Accuracy Control**, PT. PAL INDONESIA

PT PAL Indonesia (Persero), **Standar Kualitas Bangunan Kapal (Bagian Lambung)** Volume I, Nopember 1991

Santini, DrIng, Fweldl: paper **Quality Standards and Quality Control in Shipbuilding**; a Joint Task of Shipyard and Classification Society

Soejitno : Diktat Kuliah **Teknik Produksi Kapal**

Storch, Richard Lee, at. Ali,"**Ship Production**", Cornel Maritime Press,Inc,Maryland USA,1988.

Vaughan, Roger : **Productivity in Shipbuilding**, North East Coast Institution of Engineers and Shipbuilders Newcastle upon Tyne NE24he, 1983



# LAMPIRAN

## Listing Program

```
uses
  Forms,
  TedyMenu in 'TedyMenu.pas' {FormUtama};

{$R *.RES}

begin
  Application.Initialize;
  Application.Title := 'V-Ins 2.1';
  Application.CreateForm(TFormUtama, FormUtama);
  Application.Run;
end.
```

### unit TDFixer;

interface

```
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls,
  Forms, Dialogs,
  Db, DBTables, StdCtrls, DBCtrls, ExtCtrls;
```

type

```
TFormJunk = class(TForm)
  Bevel1: TBevel;
  Label1: TLabel;
  DBMemo1: TDBMemo;
  Table1: TTable;
  DataSource1: TDataSource;
  Button1: TButton;
  Bevel2: TBevel;
  Bevel3: TBevel;
  Image1: TImage;
  Bevel4: TBevel;
  Label2: TLabel;
  procedure Button1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
```

```
var
  FormJunk: TFormJunk;
```

implementation

```
{$R *.DFM}
```

```
procedure TFormJunk.Button1Click(Sender: TObject);
begin
  Close;
end;

end.
```

### unit TD Menu;

interface

```
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls,
  Forms, Dialogs,
  Menus, ComCtrls, DB, StdCtrls, ExtCtrls, Buttons, ToolWin;
```

type

```
TFormUtama = class(TForm)
  MainMenu1: TMainMenu;
  File1: TMenuItem;
  Exit1: TMenuItem;
  Help1: TMenuItem;
  About1: TMenuItem;
  StatusBar1: TStatusBar;
  ChangeDirectory1: TMenuItem;
  EntriData1: TMenuItem;
  RadioGroup1: TRadioGroup;
  Panel1: TPanel;
  BitBtn1: TBitBtn;
  CoolBar1: TCoolBar;
  SpeedButton1: TSpeedButton;
  SpeedButton2: TSpeedButton;
  CetakLaporan1: TMenuItem;
  SpeedButton3: TSpeedButton;
  DateTimePicker1: TDateTimePicker;
  procedure Exit1Click(Sender: TObject);
  procedure FormCreate(Sender: TObject);

  procedure ChangeDirectory1Click(Sender: TObject);
  procedure EntriData1Click(Sender: TObject);
  procedure FormResize(Sender: TObject);
  procedure BitBtn1Click(Sender: TObject);
  procedure SpeedButton1Click(Sender: TObject);
  procedure SpeedButton2Click(Sender: TObject);
  procedure About1Click(Sender: TObject);
  procedure CetakLaporan1Click(Sender: TObject);
  procedure SpeedButton3Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
  procedure DisplayHints(Sender: TObject);

protected
  procedure EvGetMinMaxInfo(var M: TWMGetMinMaxInfo);
  message WM_GETMINMAXINFO;
end;
```

var

```
FormUtama : TFormUtama;
d1,d2 : integer;
StringList : TStringList;
const
  tinggi = 360;
  lebar = 445;
implementation
```

```
uses inti_8, BetulDB, Suwun, CetakMas;
```

```
{$R *.DFM}
```

```
procedure TFormUtama.EvGetMinMaxInfo(var M:
```

```

TWMGetMinMaxInfo);
begin
    inherited;

    M.MinMaxInfo^.ptMinTrackSize.x:= lebar-100;
    M.MinMaxInfo^.ptMaxTrackSize.x:= lebar+100;
    M.MinMaxInfo^.ptMinTrackSize.y:= tinggi-50;
    M.MinMaxInfo^.ptMaxTrackSize.y:= tinggi+50;

    M.MinMaxInfo^.ptMaxSize.x:= 80;
    M.MinMaxInfo^.ptMaxSize.y:= 600;
    M.MinMaxInfo^.ptMaxPosition.x:= 20;
    M.MinMaxInfo^.ptMaxPosition.y:= 20;

end;

procedure CallerDLL(NamaFile,NamaProc,NamaKendali :
String);
type
    TProTestDLL = procedure(NamaSopir:String);

var
    ProTestDLL    : TProTestDLL;
    ABuffer,Buffer1 : array[0..255] of Char;
    libHandle      : THandle;
begin

    StrPCopy(Buffer1,NamaFile);
    libHandle:= LoadLibrary(Buffer1);
    if (libHandle = 0) then
        begin
            MessageDlg('Can not find the DLLs',mtError,[mbOk],0);
            exit;
        end;

    StrPCopy(ABuffer,NamaProc);

    @ProTestDLL:= GetProcAddress(libHandle,ABuffer);
    ProTestDLL(NamaKendali);

    FreeLibrary(libHandle);
end;

procedure TFormUtama.DisplayHints(Sender : TObject);
begin
    FormUtama.StatusBar1.SimpleText:=
    GetLongHint(Application.Hint);
end;

procedure TFormUtama.Exit1Click(Sender: TObject);
begin
    Close;
end;

procedure TFormUtama.FormCreate(Sender: TObject);
begin
    Application.OnHint:= DisplayHints;
    CheckState:= 1;
    CurrentDir := GetCurrentDir;
    ActDir:= CurrentDir;

    RadioGroup1.Items.Add('Fabrikasi');
    RadioGroup1.Items.Add('Sub Assembly');
    RadioGroup1.Items.Add('Assembly');
    RadioGroup1.Items.Add('Erection');
    RadioGroup1.ItemIndex:= 0;
    DateTimePicker1.Date:= Date;
end;

```

```

procedure TFormUtama.ChangeDirectory1Click(Sender:
TObject);
begin
    FormDrive:= TFormDrive.Create(Self);
    FormDrive.ShowModal;
    FormDrive.Free;
    ActDir:= CurrentDir;
end;

procedure TFormUtama.EntriData1Click(Sender: TObject);
begin
    CheckState:= 1;
    FormBetulin := TFormBetulin.Create(Self);
    with FormBetulin do
        begin
            try
                Table1.DatabaseName:= ActDir;
                Table1.TableName:= 'Baikin.DB';
                Table1.Active:= True;
            except
                on EDatabaseError do
                    begin
                        Application.MessageBox('Drive salah..!', 'Error', MB_OK or
MB_ICONERROR);
                        Table1.Active:= False;
                        CheckState:= 0;
                        FormBetulin.Free;
                        Exit;
                    end;
                end;
        end;

        if CheckState <> 0 then
            begin
                FormBetulin.ShowModal;
                FormBetulin.Free;
            end
            else FormBetulin.Free;
        end;
    end;

procedure TFormUtama.FormResize(Sender: TObject);
begin
    SendMessage(Handle, WM_ERASEBKGD, Canvas.Handle, 0);
end;

procedure TFormUtama.BitBtn1Click(Sender: TObject);
begin
    case RadioGroup1.ItemIndex of
        0 : CallerDLL('REPAIRS.DLL','FabrikasiCek',ActDir);
        1 : CallerDLL('REPAIRS.DLL','SubAssyCek',ActDir);
        2 : CallerDLL('REPAIRS.DLL','AssemblyCek',ActDir);
        3 : CallerDLL('REPAIRS.DLL','ErectionCek',ActDir);
    end;
end;

procedure TFormUtama.SpeedButton1Click(Sender: TObject);
begin
    FormUtama.ChangeDirectory1Click(Sender);
end;

procedure TFormUtama.SpeedButton2Click(Sender: TObject);
begin
    FormUtama.EntriData1Click(Sender);
end;

procedure TFormUtama.About1Click(Sender: TObject);
begin
    FormSuwun:= TFormSuwun.Create(Self);
    FormSuwun.ShowModal;
    FormSuwun.Free;
end;

```



```

end;

procedure TFormUtama.CetakLaporan1Click(Sender: TObject);
begin
    FormCetakData := TFormCetakData.Create(Self);
    FormCetakData.ShowModal;
    FormCetakData.Free;
end;

procedure TFormUtama.SpeedButton3Click(Sender: TObject);
begin
    FormUtama.CetakLaporan1Click(Sender);
end;

end.

unit Fabrik;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls,
    Forms, Dialogs,
    StdCtrls, ExtCtrls;

type
    TFormFabrikasi = class(TForm)
        Bevel1: TBevel;
        RadioGroup1: TRadioGroup;
        Bevel2: TBevel;
        Bevel3: TBevel;
        Button1: TButton;
        Button2: TButton;
        procedure Button1Click(Sender: TObject);
        procedure Button2Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

procedure FabrikasiCek(NamaDir : String);export;

implementation

{$R *.DFM}
uses inti_8;

var
    FormFabrikasi : TFormFabrikasi;
    NamaDrive : String;

procedure CallingDLL(NamaFile, NamaProc, NamaKendali :
String);
type
    TProTestDLL = procedure(NamaSopir:String);

var
    ProTestDLL : TProTestDLL;
    ABuffer, Buffer1 : array[0..255] of Char;
    libHandle : THandle;
begin
    StrPCopy(Buffer1, NamaFile);
    libHandle := LoadLibrary(Buffer1);
    if (libHandle = 0) then
        begin
            MessageDlg('Can not find the DLLs', mtError, [mbOk], 0);
            exit;
        end;

```

```

        end;

        StrPCopy(ABuffer, NamaProc);

        @ProTestDLL := GetProcAddress(libHandle, ABuffer);
        ProTestDLL(NamaKendali);

        FreeLibrary(libHandle);
    end;

procedure FabrikasiCek(NamaDir : String);export;

begin
    NamaDrive := NamaDir;
    FormFabrikasi := TFormFabrikasi.Create(nil);
    with FormFabrikasi do
        begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('Flange memanjang');
            RadioGroup1.Items.Add('Bracket flange ');
            RadioGroup1.Items.Add('Mal untuk lengkungan');
            RadioGroup1.Items.Add('Sudut dan built-up pelat');
            RadioGroup1.Items.Add('Pelat');
            RadioGroup1.Items.Add('Metode line heating');

            RadioGroup1.ItemIndex := 0;
        end;
        FormFabrikasi.ShowModal;
        FormFabrikasi.Free;
    end;

    procedure TFormFabrikasi.Button1Click(Sender: TObject);
    begin
        Close;
    end;

    procedure TFormFabrikasi.Button2Click(Sender: TObject);
    begin
        case RadioGroup1.ItemIndex of
            0 : CallingDLL('FAB_DB.DLL', 'Periksa1', NamaDrive);
            1 : CallingDLL('FAB_DB.DLL', 'Periksa2', NamaDrive);
            2 : CallingDLL('FAB_DB.DLL', 'Periksa3', NamaDrive);
            3 : CallingDLL('FAB_DB.DLL', 'Periksa4', NamaDrive);
            4 : CallingDLL('FAB_DB.DLL', 'Periksa5', NamaDrive);
            5 : CallingDLL('FAB_DB.DLL', 'Periksa6', NamaDrive);
        end;
    end;

end.

Library FAB_DB;

uses
    BraDB in 'BraDB.pas' {FormKe2},
    Sudut in 'Sudut.pas' {FormKe4},
    AngDB in 'AngDB.pas' {FormKe3},
    HeatDB in 'HeatDB.pas' {FormKe6},
    LFDB in 'LFDB.pas' {FormKe1},
    PelatDB in 'PelatDB.pas' {FormKe5};
{$R *.RES}

exports Periksa1 index 1;
exports Periksa2 index 2;
exports periksa3 index 3;
exports Periksa4 index 4;
exports Periksa5 index 5;
exports Periksa6 index 6;
begin
end.

```



### Library SAS\_DB;

```
uses
  LengSADB in 'LengSADB.pas' {FormKe8},
  CurvedDB in 'CurvedDB.pas' {FormKe10},
  PBlockDB in 'PBlockDB.pas' {FormKe9},
  RataAssDB in 'RataAssDB.pas' {FormKe7};
{$R *.RES}
```

```
exports Periksa7    index 1;
exports Periksa8    index 2;
exports periksa9    index 3;
exports Periksa10   index 4;
```

```
begin
```

```
end.
```

### Library ASS\_DB;

```
uses
  AssGapDB in 'AssGapDB.pas' {FormKe12},
  AssDefDB in 'AssDefDB.pas' {FormKe13},
  AssMADB in 'AssMADB.pas' {FormKe11};
{$R *.RES}
```

```
exports Periksa11   index 1;
exports Periksa12   index 2;
exports Periksa13   index 3;
```

```
begin
```

```
end.
```

### Library ERECT\_DB;

```
uses
  EGapDB in 'EGapDB.pas' {FormKe15},
  EMADB in 'EMADB.pas' {FormKe14};
{$R *.RES}
```

```
exports Periksa14   index 1;
exports Periksa15   index 2;
```

```
begin
```

```
end.
```

### unit AssDefDB;

```
interface
```

```
uses
```

```
  Windows, Messages, SysUtils, Classes, Graphics, Controls,
  Forms, Dialogs,
  StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;
```

```
type
```

```
TFormKe13 = class(TForm)
  Bevel1: TBevel;
  Label1: TLabel;
  ComboBox1: TComboBox;
  Bevel2: TBevel;
  RadioGroup1: TRadioGroup;
  Bevel5: TBevel;
  Button1: TButton;
  Label3: TLabel;
  Bevel6: TBevel;
  Edit1: TEdit;
```

```
  Bevel3: TBevel;
  Button2: TButton;
  procedure Button1Click(Sender: TObject);
  procedure ComboBox1Change(Sender: TObject);
  procedure ComboBox1Exit(Sender: TObject);
  procedure RadioGroup1Exit(Sender: TObject);
  procedure RadioGroup1Click(Sender: TObject);
  procedure Button2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
```

```
procedure Periksa13(NamaDir:String); export;
```

```
implementation
```

```
uses TDFixer;
{$R *.DFM}
```

```
var
```

```
  FormKe13: TFormKe13;
  SearchCode,NamaDrive : String;
  Checking : integer;
```

```
procedure Periksa13(NamaDir:String); export;
```

```
begin
```

```
  NamaDrive:= NamaDir;
  FormKe13:= TFormKe13.Create(nil);
  with FormKe13 do
  begin
    ComboBox1.Items.Add('Pelat kulit lambung : sisi paralel');
    ComboBox1.Items.Add('Pelat kulit lambung : paralel dasar');
    ComboBox1.Items.Add('Pelat kulit lambung : Haluan dan
Buritan');
    ComboBox1.Items.Add('Pelat Double bottom');
    ComboBox1.Items.Add('Dinding Sekat memanjang');
    ComboBox1.Items.Add('Dinding Sekat melintang');
    ComboBox1.Items.Add('Dinding sekat pemisah');
    ComboBox1.Items.Add('Geladak kekuatan : antara 0.6L
paralel');
    ComboBox1.Items.Add('Geladak kekuatan : Haluan dan
Buritan');
    ComboBox1.Items.Add('Geladak kekuatan : Bag. tertutup');
    ComboBox1.Items.Add('Geladak kedua : Bag. terbuka');
    ComboBox1.Items.Add('Geladak kedua : Bag. tertutup');
    ComboBox1.Items.Add('Bagian dalam lain : Web dari girder');
    ComboBox1.Items.Add('Bagian dalam lain : Transversal');
    ComboBox1.Items.Add('Floor dan Penumpu pada Double
bottom');
    ComboBox1.ItemIndex:= 0;
    RadioGroup1.Visible:= False;
    RadioGroup1.ItemIndex:= 0;
    Edit1.Text:= '';
  end;
```

```
  FormKe13.ShowModal;
```

```
  FormKe13.Free;
```

```
end;
```

```
procedure TFormKe13.Button1Click(Sender: TObject);
```

```
begin
```

```
  Close;
```

```
end;
```

```

procedure TFormKe13.ComboBox1Change(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    RadioGroup1.Visible:= True;
    case ComboBox1.ItemIndex of
        0 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 4.00 mm');
            RadioGroup1.Items.Add('> 4.00 mm');
            end;

        1 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 4.00 mm');
            RadioGroup1.Items.Add('> 4.00 mm');
            end;

        2 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 5.00 mm');
            RadioGroup1.Items.Add('> 5.00 mm');
            end;

        3 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 4.00 mm');
            RadioGroup1.Items.Add('> 4.00 mm');
            end;

        4 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 6.00 mm');
            RadioGroup1.Items.Add('> 6.00 mm');
            end;

        5 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 6.00 mm');
            RadioGroup1.Items.Add('> 6.00 mm');
            end;

        6 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 6.00 mm');
            RadioGroup1.Items.Add('> 6.00 mm');
            end;

        7 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 4.00 mm');
            RadioGroup1.Items.Add('> 4.00 mm');
            end;

        8 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 6.00 mm');
            RadioGroup1.Items.Add('> 6.00 mm');
            end;

        9 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 7.00 mm');
            RadioGroup1.Items.Add('> 7.00 mm');
            end;

        10 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 6.00 mm');

```

```

            RadioGroup1.Items.Add('> 6.00 mm');
            end;

        11 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 7.00 mm');
            RadioGroup1.Items.Add('> 7.00 mm');
            end;

        12 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 5.00 mm');
            RadioGroup1.Items.Add('> 5.00 mm');
            end;

        13 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 6.00 mm');
            RadioGroup1.Items.Add('> 6.00 mm');
            end;

        14 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 6.00 mm');
            RadioGroup1.Items.Add('> 6.00 mm');
            end;

    end;{ of case }

    RadioGroup1.ItemIndex:= 0;
    Str1:= 'ASS';
    Str2:= '03';
    if (ComboBox1.ItemIndex <= 8) then Str3:=
'0'+IntToStr(ComboBox1.ItemIndex+1)
    else Str3:= IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe13.ComboBox1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    RadioGroup1.ItemIndex:= 0;
    Str:= '%s-%s-%s-%s';
    Str1:= 'ASS';
    Str2:= '03';
    if (ComboBox1.ItemIndex <= 8) then Str3:=
'0'+IntToStr(ComboBox1.ItemIndex+1)
    else Str3:= IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe13.RadioGroup1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'ASS';
    Str2:= '03';
    if (ComboBox1.ItemIndex <= 8) then Str3:=
'0'+IntToStr(ComboBox1.ItemIndex+1)
    else Str3:= IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

```



```

procedure TFormKe13.RadioGroup1Click(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'ASS';
    Str2:= '03';
    if (ComboBox1.ItemIndex <= 8) then Str3:=
'0'+IntToStr(ComboBox1.ItemIndex+1)
    else Str3:= IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe13.Button2Click(Sender: TObject);
label 100 ;
begin
    Checking := 1;
    FormJunk:= TFormJunk.Create(Self);
    with FormJunk do
        begin
            try
                Table1.DatabaseName:= NamaDrive;
                Table1.TableName:= 'Fixer.db';
                DataSource1.DataSet:= Table1;
                DBMemo1.DataSource:= DataSource1;
                Table1.Active:= True;
            except
                on EDatabaseError do
                    begin
                        Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
                        Table1.Active:= False;
                        Checking:= 0;
                        FormJunk.Free ;
                        Exit;
                    end;
                end;

            if Checking <> 0 then
                begin
                    Table1.IndexName:= '';
                    if(not Table1.FindKey([ SearchCode])) then
                        begin
                            Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
                            FormJunk.Free;
                            Exit;
                        end
                    else
                        begin
                            FormJunk.ShowModal;
                            FormJunk.Free;
                        end;
                    end;
                end;
            end;
        end;
end;
end.

```

#### unit AssGapDB;

#### interface

#### uses

Windows, Messages, SysUtils, Classes, Graphics, Controls,  
Forms, Dialogs,  
StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;

#### type

```

TFormKe12 = class(TForm)
    Bevel1: TBevel;
    Label1: TLabel;
    ComboBox1: TComboBox;
    Bevel2: TBevel;
    RadioGroup1: TRadioGroup;
    Bevel5: TBevel;
    Button1: TButton;
    Label3: TLabel;
    Bevel6: TBevel;
    Edit1: TEdit;
    Bevel3: TBevel;
    Button2: TButton;
    procedure Button1Click(Sender: TObject);
    procedure ComboBox1Change(Sender: TObject);
    procedure ComboBox1Exit(Sender: TObject);
    procedure RadioGroup1Exit(Sender: TObject);
    procedure RadioGroup1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

procedure Periksa12(NamaDir:String); export;

implementation

uses TDFixer;
{$R *.DFM}

var
    FormKe12: TFormKe12;
    SearchCode,NamaDrive : String;
    Checking : integer;

procedure Periksa12(NamaDir:String); export;
begin
    NamaDrive:= NamaDir;
    FormKe12:= TFormKe12.Create(nil);
    with FormKe12 do
        begin
            ComboBox1.Items.Add('Gap antara pelat dan penguat');
            ComboBox1.Items.Add('Gap antara pot.terusan dg.pelat kedap');
            ComboBox1.Items.Add('Preweld Gap pada pengelasan Fillet');
            ComboBox1.Items.Add('Preweld Gap pada pengelasan Butt
Manual');
            ComboBox1.Items.Add('Preweld Gap pada pengelasan Butt
Automatik');
            ComboBox1.ItemIndex:= 0;
            RadioGroup1.Visible:= False;
            RadioGroup1.ItemIndex:= 0;
            Edit1.Text:= ' ';
        end;

        FormKe12.ShowModal;
        FormKe12.Free;
    end;
end;

```

```

procedure TFormKe12.Button1Click(Sender: TObject);
begin
    Close;
end;

```



```

procedure TFormKe12.ComboBox1Change(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:='%s-%s-%s-%s';
    RadioGroup1.Visible:= True;
    case ComboBox1.ItemIndex of
        0 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 0.00 mm');
            RadioGroup1.Items.Add('> 0.00 mm');
            end;

        1 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 0.00 mm');
            RadioGroup1.Items.Add('> 0.00 mm');
            end;

        2 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 0.00 mm');
            RadioGroup1.Items.Add('> 0.00 mm');
            end;

        3 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 0.00 mm');
            RadioGroup1.Items.Add('> 0.00 mm');
            end;

        4 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 0.00 mm');
            RadioGroup1.Items.Add('> 0.00 mm');
            end;

    end;{ of case }
    RadioGroup1.ItemIndex:= 0;
    Str1:= 'ASS';
    Str2:= '02';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= '0';
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

```

```

procedure TFormKe12.ComboBox1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    RadioGroup1.ItemIndex:= 0;
    Str:='%s-%s-%s-%s';
    Str1:= 'ASS';
    Str2:= '02';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= '0';
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

```

```

procedure TFormKe12.RadioGroup1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:='%s-%s-%s-%s';
    Str1:= 'ASS';
    Str2:= '02';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= '0';
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);

```

```

    Edit1.Text:= SearchCode;
end;

```

```

procedure TFormKe12.RadioGroup1Click(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:='%s-%s-%s-%s';
    Str1:= 'ASS';
    Str2:= '02';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= '0';
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

```

```

procedure TFormKe12.Button2Click(Sender: TObject);
label 100 ;
begin
    Checking := 1;
    FormJunk:= TFormJunk.Create(Self);
    with FormJunk do
        begin
            try
                Table1.DatabaseName:= NamaDrive;
                Table1.TableName:= 'Fixer.db';
                DataSource1.DataSet:= Table1;
                DBMemo1.DataSource:= DataSource1;
                Table1.Active:= True;
            except
                on EDatabaseError do
                    begin
                        Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
                        Table1.Active:= False;
                        Checking:= 0;
                        FormJunk.Free ;
                        Exit;
                    end;
                end;
        end;

```

```

        if Checking <> 0 then
            begin
                Table1.IndexName:= '';
                if(not Table1.FindKey([SearchCode])) then
                    begin
                        Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
                        FormJunk.Free;
                        Exit;
                    end
                else
                    begin
                        FormJunk.ShowModal;
                        FormJunk.Free;
                    end;
                end;
            end;
        end;

```

```

end;

```

```

end.

```

**unit AssMADB;**

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls,  
Forms, Dialogs,  
StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;

type

TFormKe11 = class(TForm)  
  Bevel1: TBevel;  
  Label1: TLabel;  
  ComboBox1: TComboBox;  
  Bevel2: TBevel;  
  RadioGroup1: TRadioGroup;  
  Bevel5: TBevel;  
  Button1: TButton;  
  Label3: TLabel;  
  Bevel6: TBevel;  
  Edit1: TEdit;  
  Bevel3: TBevel;  
  Button2: TButton;  
  procedure Button1Click(Sender: TObject);  
  procedure ComboBox1Change(Sender: TObject);  
  procedure ComboBox1Exit(Sender: TObject);  
  procedure RadioGroup1Exit(Sender: TObject);  
  procedure RadioGroup1Click(Sender: TObject);  
  procedure Button2Click(Sender: TObject);  
private  
  { Private declarations }  
public  
  { Public declarations }  
end;

procedure Periksa11>NamaDir:String); export;

implementation

uses TDFixer;  
{ \$R \*.DFM }

var

FormKe11: TFormKe11;  
SearchCode,>NamaDrive : String;  
Checking : integer;

procedure Periksa11>NamaDir:String); export;

begin  
  >NamaDrive:=>NamaDir;  
  FormKe11:= TFormKe11.Create(nil);  
  with FormKe11 do  
  begin  
    ComboBox1.Items.Add('Kelurusan sambungan Fillet');  
    ComboBox1.Items.Add('Kelurusan sambungan Butt');  
    ComboBox1.Items.Add('Beda antara Balok dan Frame');  
    ComboBox1.ItemIndex:= 0;  
    RadioGroup1.Visible:= False;  
    RadioGroup1.ItemIndex:= 0;  
    Edit1.Text:= ' ';

  end;

  FormKe11.ShowModal;

  FormKe11.Free;

end;  
  
procedure TFormKe11.Button1Click(Sender: TObject);  
begin

  Close;  
end;

procedure TFormKe11.ComboBox1Change(Sender: TObject);  
var Str,Str1,

    Str2,Str3,Str4 : String;

begin

  Str:= '%s-%s-%s-%s';  
  RadioGroup1.Visible:= True;  
  case ComboBox1.ItemIndex of

    0 : begin  
      RadioGroup1.Items.Clear;  
      RadioGroup1.Items.Add('<= 0.00 mm');  
      RadioGroup1.Items.Add('> 0.00 mm');  
    end;

    1 : begin  
      RadioGroup1.Items.Clear;  
      RadioGroup1.Items.Add('<= 0.00 mm');  
      RadioGroup1.Items.Add('> 0.00 mm');  
    end;

    2 : begin  
      RadioGroup1.Items.Clear;  
      RadioGroup1.Items.Add('<= 0.00 mm');  
      RadioGroup1.Items.Add('> 0.00 mm');  
    end;

  end; { of case }

  RadioGroup1.ItemIndex:= 0;  
  Str1:= 'ASS';  
  Str2:= '01';  
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);  
  Str4:= '0';  
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);  
  Edit1.Text:= SearchCode;  
end;

procedure TFormKe11.ComboBox1Exit(Sender: TObject);

var Str,Str1,  
    Str2,Str3,Str4 : String;  
begin  
  RadioGroup1.ItemIndex:= 0;  
  Str:= '%s-%s-%s-%s';  
  Str1:= 'ASS';  
  Str2:= '01';  
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);  
  Str4:= '0';  
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);  
  Edit1.Text:= SearchCode;  
end;

procedure TFormKe11.RadioGroup1Exit(Sender: TObject);

var Str,Str1,  
    Str2,Str3,Str4 : String;  
begin  
  Str:= '%s-%s-%s-%s';  
  Str1:= 'ASS';  
  Str2:= '01';  
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);  
  Str4:= '0';  
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);  
  Edit1.Text:= SearchCode;  
end;

procedure TFormKe11.RadioGroup1Click(Sender: TObject);

var Str,Str1,  
    Str2,Str3,Str4 : String;  
begin  
  Str:= '%s-%s-%s-%s';  
  Str1:= 'ASS';



```

Str2:= '01';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= '0';
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe11.Button2Click(Sender: TObject);
label 100 ;
begin
    Checking := 1;
    FormJunk:= TFormJunk.Create(Self);
    with FormJunk do
    begin
        try
            Table1.DatabaseName:= NamaDrive;
            Table1.TableName:= 'Fixer.db';
            DataSource1.DataSet:= Table1;
            DBMemo1.DataSource:= DataSource1;
            Table1.Active:= True;
        except
            on EDatabaseError do
            begin
                Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
                Table1.Active:= False;
                Checking:= 0;
                FormJunk.Free ;
                Exit;
            end;
        end;

        if Checking <> 0 then
        begin
            Table1.IndexName:= '';
            if(not Table1.FindKey([SearchCode])) then
            begin
                Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
                FormJunk.Free;
                Exit;
            end
        else
        begin
            FormJunk.ShowModal;
            FormJunk.Free;
        end;
        end;
    end;
end;

end;

end.

unit AssMas;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls,
    Forms, Dialogs,
    StdCtrls, ExtCtrls;

type
    TFormAssembly = class(TForm)
        Bevel1: TBevel;
        RadioGroup1: TRadioGroup;
        Bevel2: TBevel;
        Bevel3: TBevel;
        Button1: TButton;

```

```

        Button2: TButton;
        procedure Button1Click(Sender: TObject);
        procedure Button2Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

procedure AssemblyCek(NamaDir : String);export;

implementation

{$R *.DFM}
uses inti_8;

var
    FormAssembly : TFormAssembly;
    NamaDrive : String;

procedure CallingDLL(NamaFile,NamaProc,NamaKendali :
String);
type
    TProTestDLL = procedure(NamaSopir:String);

var
    ProTestDLL : TProTestDLL;
    ABuffer,Buffer1 : array[0..255] of Char;
    libHandle : THandle;
begin
    StrPCopy(Buffer1,NamaFile);
    libHandle:= LoadLibrary(Buffer1);
    if (libHandle = 0) then
    begin
        MessageDlg('Can not find the DLLs',mtError,[mbOk],0);
        exit;
    end;

    StrPCopy(ABuffer,NamaProc);

    @ProTestDLL:= GetProcAddress(libHandle,ABuffer);
    ProTestDLL(NamaKendali);

    FreeLibrary(libHandle);
end;

procedure AssemblyCek(NamaDir : String);export;

begin
    NamaDrive := NamaDir;
    FormAssembly := TFormAssembly.Create(nil);
    with FormAssembly do
    begin
        RadioGroup1.Items.Clear;
        RadioGroup1.Items.Add('Mis-alignment');
        RadioGroup1.Items.Add('Celah/Gap');
        RadioGroup1.Items.Add('Deformasi');
        RadioGroup1.ItemIndex:= 0;
    end;
    FormAssembly.ShowModal;
    FormAssembly.Free;
end;

procedure TFormAssembly.Button1Click(Sender: TObject);
begin
    Close;
end;

```



```

procedure TFormAssembly.Button2Click(Sender: TObject);
begin
  case RadioGroup1.ItemIndex of
    0 : CallingDLL('ASS_DB.DLL','Periksa11',NamaDrive);
    1 : CallingDLL('ASS_DB.DLL','Periksa12',NamaDrive);
    2 : CallingDLL('ASS_DB.DLL','Periksa13',NamaDrive);
  end;
end;

end.

```

# unit BetulDB;

```

interface

```

```

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls,
  Forms, Dialogs,
  StdCtrls, ExtCtrls, DBCtrls, Mask, Db, DBTables;

```

```

type

```

```

TFormBetulin = class(TForm)
  Bevel1: TBevel;
  Bevel2: TBevel;
  DBMemo1: TDBMemo;
  Label4: TLabel;
  DBNavigator1: TDBNavigator;
  Bevel3: TBevel;
  Button1: TButton;
  Table1: TTable;
  DataSource1: TDataSource;
  Label5: TLabel;
  DBEdit4: TDBEdit;
  Table1DEFCode: TStringField;
  Table1SCode: TStringField;
  Table1LCode: TStringField;
  Table1ICode: TStringField;
  Table1ACode: TStringField;
  Table1FixingDef: TMemofield;
  Label1: TLabel;
  Label2: TLabel;
  DBEdit1: TDBEdit;
  DBEdit2: TDBEdit;
  Label3: TLabel;
  Label6: TLabel;
  Label7: TLabel;
  Label8: TLabel;
  Label9: TLabel;
  DBEdit3: TDBEdit;
  DBEdit5: TDBEdit;
  DBEdit6: TDBEdit;
  DBEdit7: TDBEdit;
  ComboBox1: TComboBox;
  Table1Stage: TStringField;
  Table1Location: TStringField;
  Table1Jenis: TStringField;
  Table1Batas: TStringField;
  Bevel4: TBevel;
  DBMemo2: TDBMemo;
  Label10: TLabel;
  DBEdit8: TDBEdit;
  Label11: TLabel;
  Bevel5: TBevel;
  Image1: TImage;
  Label12: TLabel;
  Table1CommonDef: TMemofield;
  Table1StandDef: TStringField;
  procedure Button1Click(Sender: TObject);
  procedure DBEdit1Exit(Sender: TObject);

```

```

  procedure DBEdit21Exit(Sender: TObject);
  procedure DBEdit31Exit(Sender: TObject);
  procedure FormCreate(Sender: TObject);
  procedure DBEdit5Exit(Sender: TObject);
  procedure Table1PostError(DataSet: TDataSet; E:
  EDatabaseError;
    var Action: TDataAction);
  procedure ComboBox1Change(Sender: TObject);

  procedure ComboBox1Exit(Sender: TObject);
  procedure DBEdit6Exit(Sender: TObject);
  procedure DBEdit7Exit(Sender: TObject);
  procedure Table1AfterScroll(DataSet: TDataSet);
private
  { Private declarations }
public
  { Public declarations }

```

```

protected

```

```

  procedure EvGetMinMaxInfo(var M: TWMMGetMinMaxInfo);
  message WM_GETMINMAXINFO;

```

```

end;

```

```

var
  FormBetulin: TFormBetulin;

```

```

const

```

```

  eKeyViol = 9729;
  eRequiredFieldMissing = 9732;
  Lebar = 565;
  Tinggi = 415;

```

```

implementation

```

```

{$R *.DFM}

```

```

procedure TFormBetulin.EvGetMinMaxInfo(var M:
TWMMGetMinMaxInfo);
begin
  inherited;

```

```

  M.MinMaxInfo^.ptMinTrackSize.x:= lebar;
  M.MinMaxInfo^.ptMaxTrackSize.x:= lebar;
  M.MinMaxInfo^.ptMinTrackSize.y:= tinggi;
  M.MinMaxInfo^.ptMaxTrackSize.y:= tinggi;

```

```

  M.MinMaxInfo^.ptMaxSize.x:= lebar+5;
  M.MinMaxInfo^.ptMaxSize.x:= tinggi+5;
  M.MinMaxInfo^.ptMaxPosition.x:= 20;
  M.MinMaxInfo^.ptMaxPosition.y:= 20;

```

```

end;

```

```

procedure TFormBetulin.Button1Click(Sender: TObject);
begin
  Close;
end;

```

```

procedure TFormBetulin.DBEdit1Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
  Table1.Edit;
  Str:='%s-%s-%s-%s';
  Str1:= Table1SCode.Value;
  Str2:= Table1LCode.Value;

```

```

Str3:= Table1ICode.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);
Table1DEFCCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.DBEdit2Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1Scode.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICode.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

Table1DEFCCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.DBEdit3Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1Scode.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICode.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

Table1DEFCCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.FormCreate(Sender: TObject);
begin
DataSource1.DataSet:= Table1;
DBNavigator1.DataSource:= DataSource1;

ComboBox1.Items.Add('Dibawah standar');
ComboBox1.Items.Add('Diatas standar');
ComboBox1.ItemIndex:= 0;
end;

procedure TFormBetulin.DBEdit5Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1Scode.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICode.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

Table1DEFCCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.Table1PostError(DataSet: TDataSet; E:
EDatabaseError;
var Action: TDataAction);
var

```

```

iDBError: Integer;
begin
if (E is EDBEngineError) then
begin
iDBError := (E as EDBEngineError).Errors[0].ErrorCode;
case iDBError of
eRequiredFieldMissing:
begin
{ Required field not provided}
Application.MessageBox('Isilah data yang
diperlukan','Information',
MB_OK or MB_ICONINFORMATION);
Abort;
end;

eKeyViol:
{The primary key is DEFCCode}
begin
Application.MessageBox('Data tidak bisa disimpan.
Terdapat ID Code yang sama','Error',
MB_OK or MB_ICONWARNING);
Abort;
end;
end;
end;
end;
end;

```

```

procedure TFormBetulin.ComboBox1Change(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1Scode.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICode.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

Table1DEFCCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

```

```

procedure TFormBetulin.ComboBox1Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1Scode.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICode.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

Table1DEFCCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

```

```

procedure TFormBetulin.DBEdit6Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1Scode.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICode.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

```

```

Table1DEFCCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);

```



```

Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.DBEdit7Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
    Table1.Edit;
    Str:= '%s-%s-%s-%s';
    Str1:= Table1Scode.Value;
    Str2:= Table1LCode.Value;
    Str3:= Table1ICode.Value;
    Str4:= IntToStr(ComboBox1.ItemIndex);

    Table1DEFCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
    Table1ACode.Value:= Str4;
    Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.Table1AfterScroll(DataSet: TDataSet);
begin
    if Table1ACode.Value = '0' then ComboBox1.ItemIndex:= 0
    else ComboBox1.ItemIndex:= 1;
end;

end.

```

# unit BraDB;

## interface

### uses

Windows, Messages, SysUtils, Classes, Graphics, Controls,  
Forms, Dialogs,  
StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;

### type

```

TFormKe2 = class(TForm)
    Bevel1: TBevel;
    Label1: TLabel;
    ComboBox1: TComboBox;
    Bevel2: TBevel;
    RadioGroup1: TRadioGroup;
    Bevel5: TBevel;
    Button1: TButton;
    Label3: TLabel;
    Bevel6: TBevel;
    Edit1: TEdit;
    Bevel3: TBevel;
    Button2: TButton;
    procedure Button1Click(Sender: TObject);
    procedure ComboBox1Change(Sender: TObject);
    procedure ComboBox1Exit(Sender: TObject);
    procedure RadioGroup1Exit(Sender: TObject);
    procedure RadioGroup1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

```

```

procedure Periksa2>NamaDir:String); export;

```

## implementation

```

uses TDFixer;
{$R *.DFM}

```

```

var
    FormKe2: TFormKe2;
    SearchCode,>NamaDrive : String;
    Checking : integer;

procedure Periksa2>NamaDir:String); export;
begin
   >NamaDrive:=>NamaDir;
    FormKe2:= TFormKe2.Create(nil);
    with FormKe2 do
    begin
        ComboBox1.Items.Add('Lebar flens');
        ComboBox1.Items.Add('Sudut antara flens dan web');
        ComboBox1.ItemIndex:= 0;
        RadioGroup1.Visible:= False;
        RadioGroup1.ItemIndex:= 0;
        Edit1.Text:= ' ';
    end;

    FormKe2.ShowModal;
    FormKe2.Free;

end;

procedure TFormKe2.Button1Click(Sender: TObject);
begin
    Close;
end;

procedure TFormKe2.ComboBox1Change(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    RadioGroup1.Visible:= True;
    case ComboBox1.ItemIndex of
        0 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 3.00 mm');
            RadioGroup1.Items.Add('> 3.00 mm');
        end;

        1 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 3.00 mm');
            RadioGroup1.Items.Add('> 3.00 mm');
        end;

    end;{ of case }
    RadioGroup1.ItemIndex:= 0;
    Str1:= 'FAB';
    Str2:= '02';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe2.ComboBox1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    RadioGroup1.ItemIndex:= 0;
    Str:= '%s-%s-%s-%s';
    Str1:= 'FAB';
    Str2:= '02';

```



```

Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe2.RadioGroup1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
Str:= '%s-%s-%s-%s';
Str1:= 'FAB';
Str2:= '02';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe2.RadioGroup1Click(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
Str:= '%s-%s-%s-%s';
Str1:= 'FAB';
Str2:= '02';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe2.Button2Click(Sender: TObject);
label 100 ;
begin
Checking := 1;
FormJunk:= TFormJunk.Create(Self);
with FormJunk do
begin
try
Table1.DatabaseName:= NamaDrive;
Table1.TableName:= 'Fixer.db';
DataSource1.DataSet:= Table1;
DBMemo1.DataSource:= DataSource1;
Table1.Active:= True;
except
on EDatabaseError do
begin
Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
Table1.Active:= False;
Checking:= 0;
FormJunk.Free ;
Exit;
end;
end;

if Checking <> 0 then
begin
Table1.IndexName:= '';
if(not Table1.FindKey([SearchCode])) then
begin
Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
FormJunk.Free;
Exit;
end
else
begin
FormJunk.ShowModal;
FormJunk.Free;

```

```

end;
end;
end;

end;

unit CurvedDB;

interface

uses
Windows, Messages, SysUtils, Classes, Graphics, Controls,
Forms, Dialogs,
StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;

type
TFormKe10 = class(TForm)
Bevel1: TBevel;
Label1: TLabel;
ComboBox1: TComboBox;
Bevel2: TBevel;
RadioGroup1: TRadioGroup;
Bevel5: TBevel;
Button1: TButton;
Label3: TLabel;
Bevel6: TBevel;
Edit1: TEdit;
Bevel3: TBevel;
Button2: TButton;
procedure Button1Click(Sender: TObject);
procedure ComboBox1Change(Sender: TObject);
procedure ComboBox1Exit(Sender: TObject);
procedure RadioGroup1Exit(Sender: TObject);
procedure RadioGroup1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
private
{ Private declarations }
public
{ Public declarations }
end;

procedure Periksa10(NamaDir:String); export;

implementation

uses TDFixer,
{$R *.DFM}

var
FormKe10: TFormKe10;
SearchCode,NamaDrive : String;
Checking : integer;

procedure Periksa10(NamaDir:String); export;
begin
NamaDrive:= NamaDir,
FormKe10:= TFormKe10.Create(nil);
with FormKe10 do
begin
ComboBox1.Items.Add('Lebar tiap panel');
ComboBox1.Items.Add('Panjang tiap panel');
ComboBox1.Items.Add('Puntiran tiap panel');
ComboBox1.Items.Add('Penyimpangan komponen dalam
thd.pelat kulit');
ComboBox1.Items.Add('Twisting');
ComboBox1.Items.Add('Penyimpangan panel thd. CL atau BL');
ComboBox1.Items.Add('Penyimpangan panel thd. Frame Line');

```

```

ComboBox1.ItemIndex:= 0;
RadioGroup1.Visible:= False;
RadioGroup1.ItemIndex:= 0;
Edit1.Text:= ' ';
end;

FormKe10.ShowModal;

FormKe10.Free;

end;

procedure TFormKe10.Button1Click(Sender: TObject);
begin
  Close;
end;

procedure TFormKe10.ComboBox1Change(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
  Str:= '%s-%s-%s-%s';
  RadioGroup1.Visible:= True;
  case ComboBox1.ItemIndex of
    0 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 4.00 mm');
      RadioGroup1.Items.Add('> 4.00 mm');
      end;
    1 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 4.00 mm');
      RadioGroup1.Items.Add('> 4.00 mm');
      end;
    2 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 10.00 mm');
      RadioGroup1.Items.Add('> 10.00 mm');
      end;
    3 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 5.00 mm');
      RadioGroup1.Items.Add('> 5.00 mm');
      end;
    4 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 15.00 mm');
      RadioGroup1.Items.Add('> 15.00 mm');
      end;
    5 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 7.00 mm');
      RadioGroup1.Items.Add('> 7.00 mm');
      end;
    6 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 7.00 mm');
      RadioGroup1.Items.Add('> 7.00 mm');
      end;
  end;{ of case }
  RadioGroup1.ItemIndex:= 0;

```

```

Str1:= 'SAS';
Str2:= '04';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

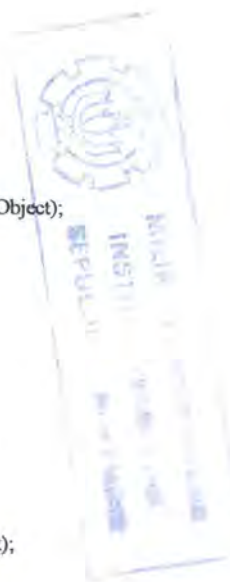
procedure TFormKe10.ComboBox1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
  RadioGroup1.ItemIndex:= 0;
  Str:= '%s-%s-%s-%s';
  Str1:= 'SAS';
  Str2:= '04';
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
  Str4:= IntToStr(RadioGroup1.ItemIndex);
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
  Edit1.Text:= SearchCode;
end;

procedure TFormKe10.RadioGroup1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
  Str:= '%s-%s-%s-%s';
  Str1:= 'SAS';
  Str2:= '04';
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
  Str4:= IntToStr(RadioGroup1.ItemIndex);
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
  Edit1.Text:= SearchCode;
end;

procedure TFormKe10.RadioGroup1Click(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
  Str:= '%s-%s-%s-%s';
  Str1:= 'SAS';
  Str2:= '04';
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
  Str4:= IntToStr(RadioGroup1.ItemIndex);
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
  Edit1.Text:= SearchCode;
end;

procedure TFormKe10.Button2Click(Sender: TObject);
label 100;
begin
  Checking := 1;
  FormJunk:= TFormJunk.Create(Self);
  with FormJunk do
    begin
      try
        Table1.DatabaseName:= NamaDrive;
        Table1.TableName:= 'Fixer.db';
        DataSource1.DataSet:= Table1;
        DBMemo1.DataSource:= DataSource1;
        Table1.Active:= True;
      except
        on EDatabaseError do
          begin
            Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
            Table1.Active:= False;
            Checking:= 0;
            FormJunk.Free ;
            Exit;
          end;

```





```

end;

if Checking <> 0 then
begin
  Table1.IndexName:='';
  if(not Table1.FindKey([SearchCode])) then
  begin
    Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
    FormJunk.Free;
    Exit;
  end
else
  begin
    FormJunk.ShowModal;
    FormJunk.Free;
  end;
end;
end;

end;

end.

```

#### unit EGapDB;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls,  
Forms, Dialogs,  
StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;

type

```

TFormKe15 = class(TForm)
  Bevel1: TBevel;
  Label1: TLabel;
  ComboBox1: TComboBox;
  Bevel2: TBevel;
  RadioGroup1: TRadioGroup;
  Bevel5: TBevel;
  Button1: TButton;
  Label3: TLabel;
  Bevel6: TBevel;
  Edit1: TEdit;
  Bevel3: TBevel;
  Button2: TButton;
  procedure Button1Click(Sender: TObject);
  procedure ComboBox1Change(Sender: TObject);
  procedure ComboBox1Exit(Sender: TObject);
  procedure RadioGroup1Exit(Sender: TObject);
  procedure RadioGroup1Click(Sender: TObject);
  procedure Button2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

```

procedure Periksa15>NamaDir:String); export;

implementation

uses TDFixer,  
{ \$R \*.DFM }

var

FormKe15: TFormKe15;  
SearchCode,>NamaDrive : String;  
Checking : integer;

procedure Periksa15>NamaDir:String); export;

begin

```

 >NamaDrive:=>NamaDir;
  FormKe15:= TFormKe15.Create(nil);
  with FormKe15 do
  begin
    ComboBox1.Items.Add('Gap antara pelat dan penguat');
    ComboBox1.Items.Add('Gap antara pot.terusan dg.pelat kedap');
    ComboBox1.Items.Add('Preweld Gap pada pengelasan Fillet');
    ComboBox1.Items.Add('Preweld Gap pada pengelasan Butt
Manual');
    ComboBox1.Items.Add('Preweld Gap pada pengelasan Butt
Automatik');
    ComboBox1.ItemIndex:= 0;
    RadioGroup1.Visible:= False;
    RadioGroup1.ItemIndex:= 0;
    Edit1.Text:= ' ';
  end;

```

```

  FormKe15.ShowModal;
  FormKe15.Free;

```

end;

procedure TFormKe15.Button1Click(Sender: TObject);

begin

```

  Close;
end;
```

procedure TFormKe15.ComboBox1Change(Sender: TObject);

var Str,Str1,

Str2,Str3,Str4 : String;

begin

```

  Str:= '%s-%s-%s-%s';
  RadioGroup1.Visible:= True;
  case ComboBox1.ItemIndex of
    0 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('c dan B <= 3.00 mm');
      RadioGroup1.Items.Add('c dan B > 0.00 mm');
    end;

```

1 : begin

```

  RadioGroup1.Items.Clear;
  RadioGroup1.Items.Add('c1 <= 3.00 mm');
  RadioGroup1.Items.Add('c1 > 3.00 mm');
end;
```

2 : begin

```

  RadioGroup1.Items.Clear;
  RadioGroup1.Items.Add('a <= 2.00 mm');
  RadioGroup1.Items.Add('a > 2.00 mm');
end;
```

3 : begin

```

  RadioGroup1.Items.Clear;
  RadioGroup1.Items.Add('2.00 <= a < 3.50 mm');
  RadioGroup1.Items.Add('a > 3.50 mm');
end;
```

4 : begin

```

  RadioGroup1.Items.Clear;
  RadioGroup1.Items.Add('0.00 <= a < 0.80 mm');
  RadioGroup1.Items.Add('a > 0.80 mm');
end;
```



```

end; { of case }
RadioGroup1.ItemIndex:= 0;
Str1:= 'ERC';
Str2:= '02';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe15.ComboBox1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    RadioGroup1.ItemIndex:= 0;
    Str:= '%s-%s-%s-%s';
    Str1:= 'ERC';
    Str2:= '02';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe15.RadioGroup1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'ERC';
    Str2:= '02';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe15.RadioGroup1Click(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'ERC';
    Str2:= '02';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe15.Button2Click(Sender: TObject);
label 100;
begin
    Checking := 1;
    FormJunk:= TFormJunk.Create(Self);
    with FormJunk do
    begin
        try
            Table1.DatabaseName:= NamaDrive;
            Table1.TableName:= 'Fixer.db';
            DataSource1.DataSet:= Table1;
            DBMemo1.DataSource:= DataSource1;
            Table1.Active:= True;
        except
            on EDatabaseError do
            begin
                Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
                Table1.Active:= False;
                Checking:= 0;
                FormJunk.Free;
            end;
        end;
    end;
end;

```

```

Exit;
end;
end;

if Checking <> 0 then
begin
    Table1.IndexName:= '';
    if(not Table1.FindKey([SearchCode])) then
    begin
        Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
        FormJunk.Free;
        Exit;
    end
    else
    begin
        FormJunk.ShowModal;
        FormJunk.Free;
    end;
end;
end;

end;

end.

unit EMADB;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls,
    Forms, Dialogs,
    StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;

type
    TFormKe14 = class(TForm)
        Bevel1: TBevel;
        Label1: TLabel;
        ComboBox1: TComboBox;
        Bevel2: TBevel;
        RadioGroup1: TRadioGroup;
        Bevel5: TBevel;
        Button1: TButton;
        Label3: TLabel;
        Bevel6: TBevel;
        Edit1: TEdit;
        Bevel3: TBevel;
        Button2: TButton;
    procedure Button1Click(Sender: TObject);
    procedure ComboBox1Change(Sender: TObject);
    procedure ComboBox1Exit(Sender: TObject);
    procedure RadioGroup1Exit(Sender: TObject);
    procedure RadioGroup1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

    procedure Periksa14(NamaDir:String); export;

implementation

uses TDFixer;
{$R *.DFM}

var
    FormKe14: TFormKe14;

```

```

SearchCode,NamaDrive : String;
Checking : integer;

procedure Periksa14(NamaDir:String); export;
begin
    NamaDrive:= NamaDir;
    FormKe14:= TFormKe14.Create(nil);
    with FormKe14 do
    begin
        ComboBox1.Items.Add('Kelurusan sambungan Fillet');
        ComboBox1.Items.Add('Kelurusan sambungan Butt');
        ComboBox1.Items.Add('Beda antara Balok dan Frame');
        ComboBox1.ItemIndex:= 0;
        RadioGroup1.Visible:= False;
        RadioGroup1.ItemIndex:= 0;
        Edit1.Text:= ' ';
    end;

    FormKe14.ShowModal;

    FormKe14.Free;

end;

procedure TFormKe14.Button1Click(Sender: TObject);
begin
    Close;
end;

procedure TFormKe14.ComboBox1Change(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    RadioGroup1.Visible:= True;
    case ComboBox1.ItemIndex of
        0 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('a <= 1/3 t');
            RadioGroup1.Items.Add('a > 1/3 t');
        end;
        1 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('a <= 0.15 t ; max. 3 mm');
            RadioGroup1.Items.Add('a > 0.15 t ; max. 3 mm');
        end;
        2 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('a <= 3.00 mm');
            RadioGroup1.Items.Add('a > 3.00 mm');
        end;
    end;{ of case }

    RadioGroup1.ItemIndex:= 0;
    Str1:= 'ERC';
    Str2:= '01';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe14.ComboBox1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    RadioGroup1.ItemIndex:= 0;

```

```

    Str:= '%s-%s-%s-%s';
    Str1:= 'ERC';
    Str2:= '01';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe14.RadioGroup1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'ERC';
    Str2:= '01';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe14.RadioGroup1Click(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'ERC';
    Str2:= '01';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe14.Button2Click(Sender: TObject);
label 100 ;
begin
    Checking := 1;
    FormJunk:= TFormJunk.Create(Self);
    with FormJunk do
    begin
        try
            Table1.DatabaseName:= NamaDrive;
            Table1.TableName:= 'Fixer.db';
            DataSource1.DataSet:= Table1;
            DBMemo1.DataSource:= DataSource1;
            Table1.Active:= True;
        except
            on EDatabaseError do
            begin
                Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
                Table1.Active:= False;
                Checking:= 0;
                FormJunk.Free ;
                Exit;
            end;
        end;

        if Checking <> 0 then
        begin
            Table1.IndexName:= '';
            if(not Table1.FindKey([SearchCode])) then
            begin
                Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
                FormJunk.Free;
                Exit;
            end
            else

```



```

begin
    FormJunk.ShowModal;
    FormJunk.Free;
end;
end;
end;

end.
unit LengSADB;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls,
    Forms, Dialogs,
    StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;

type
    TFormKe8 = class(TForm)
        Bevel1: TBevel;
        Label1: TLabel;
        ComboBox1: TComboBox;
        Bevel2: TBevel;
        RadioGroup1: TRadioGroup;
        Bevel5: TBevel;
        Button1: TButton;
        Label3: TLabel;
        Bevel6: TBevel;
        Edit1: TEdit;
        Bevel3: TBevel;
        Button2: TButton;
    procedure Button1Click(Sender: TObject);
    procedure ComboBox1Change(Sender: TObject);
    procedure ComboBox1Exit(Sender: TObject);
    procedure RadioGroup1Exit(Sender: TObject);
    procedure RadioGroup1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

procedure Periksa8>NamaDir:String); export;

implementation

uses TDFixer;
{$R *.DFM}

var
    FormKe8: TFormKe8;
    SearchCode>NamaDrive : String;
    Checking : integer;

procedure Periksa8>NamaDir:String); export;
begin
   >NamaDrive:=>NamaDir;
    FormKe8:= TFormKe8.Create(nil);
    with FormKe8 do
    begin
        ComboBox1.Items.Add('Lebar flens');
        ComboBox1.Items.Add('Panjang flens');
        ComboBox1.Items.Add('Pemuntiran / Distorsi');
        ComboBox1.Items.Add('Kesikuan');
        ComboBox1.Items.Add('Penyimpangan member thd. frame');
        ComboBox1.ItemIndex:= 0;
        RadioGroup1.Visible:= False;

```

```

        RadioGroup1.ItemIndex:= 0;
        Edit1.Text:= '';
    end;

    FormKe8.ShowModal;
    FormKe8.Free;

end;

procedure TFormKe8.Button1Click(Sender: TObject);
begin
    Close;
end;

procedure TFormKe8.ComboBox1Change(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    RadioGroup1.Visible:= True;
    case ComboBox1.ItemIndex of
        0 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 4.00 mm');
            RadioGroup1.Items.Add('> 4.00 mm');
        end;

        1 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 4.00 mm');
            RadioGroup1.Items.Add('> 4.00 mm');
        end;

        2 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 10.00 mm');
            RadioGroup1.Items.Add('> 10.00 mm');
        end;

        3 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 10.00 mm');
            RadioGroup1.Items.Add('> 10.00 mm');
        end;

        4 : begin
            RadioGroup1.Items.Clear;
            RadioGroup1.Items.Add('<= 5.00 mm');
            RadioGroup1.Items.Add('> 5.00 mm');
        end;

    end; { of case }
    RadioGroup1.ItemIndex:= 0;
    Str1:= 'SAS';
    Str2:= '02';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe8.ComboBox1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    RadioGroup1.ItemIndex:= 0;
    Str:= '%s-%s-%s-%s';
    Str1:= 'SAS';

```



```

Str2:= '02';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe8.RadioGroup1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'SAS';
    Str2:= '02';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe8.RadioGroup1Click(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'SAS';
    Str2:= '02';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe8.Button2Click(Sender: TObject);
label 100 ;
begin
    Checking := 1;
    FormJunk:= TFormJunk.Create(Self);
    with FormJunk do
        begin
            try
                Table1.DatabaseName:= NamaDrive;
                Table1.TableName:= 'Fixer.db';
                DataSource1.DataSet:= Table1;
                DBMemo1.DataSource:= DataSource1;
                Table1.Active:= True;
            except
                on EDatabaseError do
                    begin
                        Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
                        Table1.Active:= False;
                        Checking:= 0;
                        FormJunk.Free ;
                        Exit;
                    end;
                end;
            if Checking <> 0 then
                begin
                    Table1.IndexName:= '';
                    if(not Table1.FindKey([SearchCode])) then
                        begin
                            Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
                            FormJunk.Free;
                            Exit;
                        end
                    else
                        begin
                            FormJunk.ShowModal;

```

```

FormJunk.Free;
        end;
    end;
end;

end;

unit LFDB;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls,
    Forms, Dialogs,
    StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask, ComCtrls;

type
    TFormKe1 = class(TForm)
        Bevel1: TBevel;
        Label1: TLabel;
        ComboBox1: TComboBox;
        Bevel2: TBevel;
        RadioGroup1: TRadioGroup;
        Bevel5: TBevel;
        Button1: TButton;
        Label3: TLabel;
        Bevel6: TBevel;
        Edit1: TEdit;
        Bevel3: TBevel;
        Button2: TButton;
        procedure Button1Click(Sender: TObject);
        procedure ComboBox1Change(Sender: TObject);
        procedure ComboBox1Exit(Sender: TObject);
        procedure RadioGroup1Exit(Sender: TObject);
        procedure RadioGroup1Click(Sender: TObject);
        procedure Button2Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

procedure Periksa1(NamaDir:String); export;

implementation

uses TDFixer,
    {$R *.DFM};

var
    FormKe1: TFormKe1;
    SearchCode,NamaDrive : String;
    Checking : integer;

procedure Periksa1(NamaDir:String); export;
begin
    NamaDrive:= NamaDir;
    FormKe1:= TFormKe1.Create(nil);
    with FormKe1 do
        begin
            ComboBox1.Items.Add('Lebar flens');
            ComboBox1.Items.Add('Tinggi web');
            ComboBox1.Items.Add('Sudut antara flens dan web');
            ComboBox1.Items.Add('Kelengkungan flens');
            ComboBox1.Items.Add('Kelengkungan web');
            ComboBox1.ItemIndex:= 0;

```

```

RadioGroup1.Visible:= False;
RadioGroup1.ItemIndex:= 0;
Edit1.Text:= '';
end;
FormKe1.ShowModal;

FormKe1.Free;

end;

procedure TFormKe1.Button1Click(Sender: TObject);
begin
  Close;
end;

procedure TFormKe1.ComboBox1Change(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
  Str:= '%s-%s-%s-%s';
  RadioGroup1.Visible:= True;
  case ComboBox1.ItemIndex of
    0 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 3.00 mm');
      RadioGroup1.Items.Add('> 3.00 mm');
      end;

    1 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 3.00 mm ; 2.00 mm u/
peguatan');
      RadioGroup1.Items.Add('> 3.00 mm ; 2.00 mm u/
peguatan');
      end;

    2 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 2.50 %');
      RadioGroup1.Items.Add('> 2.50 %');
      end;

    3 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 10.00 mm');
      RadioGroup1.Items.Add('> 10.00 mm');
      end;

    4 : begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('<= 10.00 mm');
      RadioGroup1.Items.Add('> 10.00 mm');
      end;

  end;{ of case }
  RadioGroup1.ItemIndex:= 0;
  Str1:= 'FAB';
  Str2:= '01';
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
  Str4:= IntToStr(RadioGroup1.ItemIndex);
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
  Edit1.Text:= SearchCode;
end;

procedure TFormKe1.ComboBox1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin

```

```

RadioGroup1.ItemIndex:= 0;
Str:= '%s-%s-%s-%s';
Str1:= 'FAB';
Str2:= '01';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe1.RadioGroup1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
  Str:= '%s-%s-%s-%s';
  Str1:= 'FAB';
  Str2:= '01';
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
  Str4:= IntToStr(RadioGroup1.ItemIndex);
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
  Edit1.Text:= SearchCode;
end;

procedure TFormKe1.RadioGroup1Click(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
  Str:= '%s-%s-%s-%s';
  Str1:= 'FAB';
  Str2:= '01';
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
  Str4:= IntToStr(RadioGroup1.ItemIndex);
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
  Edit1.Text:= SearchCode;
end;

procedure TFormKe1.Button2Click(Sender: TObject);
label 100 ;
begin
  Checking := 1;
  FormJunk:= TFormJunk.Create(Self);
  with FormJunk do
    begin
      try
        Table1.DatabaseName:= NamaDrive;
        Table1.TableName:= 'Fixer.db';
        DataSource1.DataSet:= Table1;
        DBMemo1.DataSource:= DataSource1;
        Table1.Active:= True;
      except
        on EDatabaseError do
          begin
            Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
            Table1.Active:= False;
            Checking:= 0;
            FormJunk.Free ;
            Exit;
          end;
        end;
      end;

      if Checking <> 0 then
        begin
          Table1.IndexName:= '';
          if(not Table1.FindKey([SearchCode])) then
            begin
              Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
              FormJunk.Free;
              Exit;
            end
          end;
        end;
      end;
    end;
  end;
end;

```



```

else
begin
FormJunk.ShowModal;
FormJunk.Free;
end;
end;
end;

end;

end.

unit BetulDB;

interface

uses
Windows, Messages, SysUtils, Classes, Graphics, Controls,
Forms, Dialogs,
StdCtrls, ExtCtrls, DBCtrls, Mask, Db, DBTables;

type
TFormBetulin = class(TForm)
Bevel1: TBevel;
Bevel2: TBevel;
DBMemo1: TDBMemo;
Label4: TLabel;
DBNavigator1: TDBNavigator;
Bevel3: TBevel;
Button1: TButton;
Table1: TTable;
DataSource1: TDataSource;
Label5: TLabel;
DBEdit4: TDBEdit;
Table1DEFCODE: TStringField;
Table1SCODE: TStringField;
Table1LCODE: TStringField;
Table1ICODE: TStringField;
Table1ACODE: TStringField;
Table1FixingDef: TMemoField;
Label1: TLabel;
Label2: TLabel;
DBEdit1: TDBEdit;
DBEdit2: TDBEdit;
Label3: TLabel;
Label6: TLabel;
Label7: TLabel;
Label8: TLabel;
Label9: TLabel;
DBEdit3: TDBEdit;
DBEdit5: TDBEdit;
DBEdit6: TDBEdit;
DBEdit7: TDBEdit;
ComboBox1: TComboBox;
Table1Stage: TStringField;
Table1Location: TStringField;
Table1Jenis: TStringField;
Table1Batas: TStringField;
Bevel4: TBevel;
DBMemo2: TDBMemo;
Label10: TLabel;
DBEdit8: TDBEdit;
Label11: TLabel;
Bevel5: TBevel;
Image1: TImage;
Label12: TLabel;
Table1CommonDef: TMemoField;
Table1StandDef: TStringField;
procedure Button1Click(Sender: TObject);
procedure DBEdit1Exit(Sender: TObject);

```

```

procedure DBEdit2Exit(Sender: TObject);
procedure DBEdit3Exit(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure DBEdit5Exit(Sender: TObject);
procedure Table1PostError(DataSet: TDataSet; E:
EDatabaseError;
var Action: TDataAction);
procedure ComboBox1Change(Sender: TObject);

procedure ComboBox1Exit(Sender: TObject);
procedure DBEdit6Exit(Sender: TObject);
procedure DBEdit7Exit(Sender: TObject);
procedure Table1AfterScroll(DataSet: TDataSet);
private
{ Private declarations }
public
{ Public declarations }

protected

procedure EvGetMinMaxInfo(var M: TWMGetMinMaxInfo);
message WM_GETMINMAXINFO;

end;

var
FormBetulin: TFormBetulin;

const
eKeyViol = 9729;
eRequiredFieldMissing = 9732;
Lebar = 565;
Tinggi = 415;

implementation

{$R *.DFM}

procedure TFormBetulin.EvGetMinMaxInfo(var M:
TWMGetMinMaxInfo);
begin
inherited;

M.MinMaxInfo^.ptMinTrackSize.x:= lebar;
M.MinMaxInfo^.ptMaxTrackSize.x:= lebar;
M.MinMaxInfo^.ptMinTrackSize.y:= tinggi;
M.MinMaxInfo^.ptMaxTrackSize.y:= tinggi;

M.MinMaxInfo^.ptMaxSize.x:= lebar+5;
M.MinMaxInfo^.ptMaxSize.y:= tinggi+5;
M.MinMaxInfo^.ptMaxPosition.x:= 20;
M.MinMaxInfo^.ptMaxPosition.y:= 20;

end;

procedure TFormBetulin.Button1Click(Sender: TObject);
begin
Close;
end;

procedure TFormBetulin.DBEdit1Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1SCODE.Value;
Str2:= Table1LCODE.Value;

```



```

Str3:= Table1LCODE.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);
Table1DEFCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.DBEdit2Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1SCODE.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICODE.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

Table1DEFCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.DBEdit3Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1SCODE.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICODE.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

Table1DEFCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.FormCreate(Sender: TObject);
begin
DataSource1.DataSet:= Table1;
DBNavigator1.DataSource:= DataSource1;

ComboBox1.Items.Add('Dibawah standar');
ComboBox1.Items.Add('Diatas standar');
ComboBox1.ItemIndex:= 0;
end;

procedure TFormBetulin.DBEdit5Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1SCODE.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICODE.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

Table1DEFCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.Table1PostError(DataSet: TDataSet; E:
EDatabaseError;
var Action: TDataAction);
var

```

```

iDBIError: Integer;
begin
if (E is EDBEngineError) then
begin
iDBIError := (E as EDBEngineError).Errors[0].ErrorCode;
case iDBIError of
eRequiredFieldMissing:
begin
{ Required field not provided}
Application.MessageBox('Isilah data yang
diperlukan','Information',
MB_OK or MB_ICONINFORMATION);
Abort;
end;

eKeyViol:
{The primary key is DEFCode}
begin
Application.MessageBox('Data tidak bisa disimpan.
Terdapat ID Code yang sama','Error',
MB_OK or MB_ICONWARNING);
Abort;
end;
end;
end;
end;

procedure TFormBetulin.ComboBox1Change(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1SCODE.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICODE.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

Table1DEFCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.ComboBox1Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1SCODE.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICODE.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

Table1DEFCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
Table1ACode.Value:= Str4;
Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.DBEdit6Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
Table1.Edit;
Str:='%s-%s-%s-%s';
Str1:= Table1SCODE.Value;
Str2:= Table1LCODE.Value;
Str3:= Table1ICODE.Value;
Str4:= IntToStr(ComboBox1.ItemIndex);

Table1DEFCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);

```

```

    Table1ACode.Value:= Str4;
    Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.DBEdit7Exit(Sender: TObject);
var Str,Str1,Str2,Str3,Str4 : String;
begin
    Table1.Edit;
    Str:='%s-%s-%s-%s';
    Str1:= Table1Scode.Value;
    Str2:= Table1LCode.Value;
    Str3:= Table1ICode.Value;
    Str4:= IntToStr(ComboBox1.ItemIndex);

    Table1DEFCCode.Value:= Format(Str,[Str1,Str2,Str3,Str4]);
    Table1ACode.Value:= Str4;
    Table1Batas.Value:=
ComboBox1.Items[ComboBox1.ItemIndex];
end;

procedure TFormBetulin.Table1AfterScroll(DataSet: TDataSet);
begin
    if Table1ACode.Value = '0' then ComboBox1.ItemIndex:= 0
    else ComboBox1.ItemIndex:= 1;
end;

end.

```

**unit PelatDB;**

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls,  
Forms, Dialogs,  
StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;

type

TFormKe5 = class(TForm)

Bevel1: TBevel;

Label1: TLabel;

ComboBox1: TComboBox;

Bevel2: TBevel;

RadioGroup1: TRadioGroup;

Bevel5: TBevel;

Button1: TButton;

Label3: TLabel;

Bevel6: TBevel;

Edit1: TEdit;

Bevel3: TBevel;

Button2: TButton;

procedure Button1Click(Sender: TObject);

procedure ComboBox1Change(Sender: TObject);

procedure ComboBox1Exit(Sender: TObject);

procedure RadioGroup1Exit(Sender: TObject);

procedure RadioGroup1Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

procedure Periksa5>NamaDir:String); export;

implementation

uses TDFixer;

{ \$R \*.DFM }

var

FormKe5: TFormKe5;

SearchCode>NamaDrive : String;

Checking : integer;

procedure Periksa5>NamaDir:String); export;

begin

NamaDrive:=>NamaDir;

FormKe5:= TFormKe5.Create(nil);

with FormKe5 do

begin

ComboBox1.Items.Add('Tinggi gelombang pada Sekat  
gelombang');

ComboBox1.Items.Add('Lebar gelombang pada Sekat  
gelombang');

ComboBox1.Items.Add('Puncak gelombang pada Dinding  
gelombang');

ComboBox1.Items.Add('Tinggi gel. ; puncak tidak terhubung');

ComboBox1.Items.Add('Tinggi gel. ; puncak terhubung');

ComboBox1.Items.Add('Diameter pada bentuk silindrik');

ComboBox1.Items.Add('Kelengkungan memanjang thd. garis  
cek');

ComboBox1.Items.Add('Kelengkungan melintang thd. garis  
cek');

ComboBox1.Items.Add('Gap antara pelat kulit dan mal seksi');

ComboBox1.ItemIndex:= 0;

RadioGroup1.Visible:= False;

RadioGroup1.ItemIndex:= 0;

Edit1.Text:=' ';

end;

FormKe5.ShowModal;

FormKe5.Free;

end;

procedure TFormKe5.Button1Click(Sender: TObject);

begin

Close;

end;

procedure TFormKe5.ComboBox1Change(Sender: TObject);

var Str,Str1,

Str2,Str3,Str4 : String;

begin

Str:= '%s-%s-%s-%s';

RadioGroup1.Visible:= True;

case ComboBox1.ItemIndex of

0 : begin

RadioGroup1.Items.Clear;

RadioGroup1.Items.Add('<= 3.00 mm');

RadioGroup1.Items.Add('> 3.00 mm');

end;

1 : begin

RadioGroup1.Items.Clear;

RadioGroup1.Items.Add('<= 2.00 mm');

RadioGroup1.Items.Add('> 2.00 mm');

end;

2 : begin

RadioGroup1.Items.Clear;

RadioGroup1.Items.Add('<= 6.00');

RadioGroup1.Items.Add('> 6.00');

end;

3 : begin

RadioGroup1.Items.Clear;

RadioGroup1.Items.Add('<= 2.00 mm');

RadioGroup1.Items.Add('> 2.00 mm');

end;

4 : begin

RadioGroup1.Items.Clear;

RadioGroup1.Items.Add('<= 2.50 mm');

RadioGroup1.Items.Add('> 2.50 mm');

end;

5 : begin

RadioGroup1.Items.Clear;

RadioGroup1.Items.Add('<= D/200 atau 5.00 mm');

RadioGroup1.Items.Add('> D/200 atau 5.00 mm');

end;

6 : begin

RadioGroup1.Items.Clear;

RadioGroup1.Items.Add('<= 2.50 mm');

RadioGroup1.Items.Add('> 2.50 mm');

end;

7 : begin

RadioGroup1.Items.Clear;

RadioGroup1.Items.Add('<= 2.50 mm');

RadioGroup1.Items.Add('> 2.50 mm');

end;



```

8 : begin
    RadioGroup1.Items.Clear;
    RadioGroup1.Items.Add('<= 2.50 mm');
    RadioGroup1.Items.Add('> 2.50 mm');
end;

end; { of case }

RadioGroup1.ItemIndex:= 0;
Str1:= 'FAB';
Str2:= '05';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe5.ComboBox1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    RadioGroup1.ItemIndex:= 0;
    Str:= '%s-%s-%s-%s';
    Str1:= 'FAB';
    Str2:= '05';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe5.RadioGroup1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'FAB';
    Str2:= '05';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe5.RadioGroup1Click(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'FAB';
    Str2:= '05';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

procedure TFormKe5.Button2Click(Sender: TObject);
label 100 ;
begin
    Checking := 1;
    FormJunk:= TFormJunk.Create(Self);
    with FormJunk do
        begin
            try
                Table1.DatabaseName:= NamaDrive;
                Table1.TableName:= 'Fixer.db';
                DataSource1.DataSet:= Table1;
                DBMemo1.DataSource:= DataSource1;
                Table1.Active:= True;

```

```

except
on EDatabaseError do
begin
    Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
    Table1.Active:= False;
    Checking:= 0;
    FormJunk.Free ;
    Exit;
end;
end;

if Checking <> 0 then
begin
    Table1.IndexName:= '';
    if(not Table1.FindKey([SearchCode])) then
        begin
            Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
            FormJunk.Free;
            Exit;
        end
    else
        begin
            FormJunk.ShowModal;
            FormJunk.Free;
        end;
    end;
end;

end;

end.

unit Sudut;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls,
    Forms, Dialogs,
    StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;

type
    TFormKe4 = class(TForm)
        Bevel1: TBevel;
        Label1: TLabel;
        ComboBox1: TComboBox;
        Bevel2: TBevel;
        RadioGroup1: TRadioGroup;
        Bevel5: TBevel;
        Button1: TButton;
        Label3: TLabel;
        Bevel6: TBevel;
        Edit1: TEdit;
        Bevel3: TBevel;
        Button2: TButton;
    procedure Button1Click(Sender: TObject);
    procedure ComboBox1Change(Sender: TObject);
    procedure ComboBox1Exit(Sender: TObject);
    procedure RadioGroup1Exit(Sender: TObject);
    procedure RadioGroup1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

procedure Periksa4(NamaDir:String); export;

```

implementation

```
uses TDFixer;  
{ $R *.DFM }
```

```
var  
  FormKe4 : TFormKe4;  
  SearchCode, NamaDrive : String;  
  Checking : integer;
```

```
procedure Periksa4(NamaDir:String); export;  
begin  
  NamaDrive:= NamaDir;  
  FormKe4:= TFormKe4.Create(nil);  
  with FormKe4 do  
    begin  
      ComboBox1.Items.Add('Sudut stringer thd. pengukur sudut');  
      ComboBox1.Items.Add('Sudut stringer thd. mal');  
      ComboBox1.Items.Add('Lengkungan thd. garis cek');  
      ComboBox1.Items.Add('Penyimpangan bentuk');  
      ComboBox1.Items.Add('Penyimpangan sudut flens');  
      ComboBox1.Items.Add('Penyimpangan pelat face');  
      ComboBox1.ItemIndex:= 0;  
      RadioGroup1.Visible:= False;  
      RadioGroup1.ItemIndex:= 0;  
      Edit1.Text:= ' ';    end;
```

```
  FormKe4.ShowModal;
```

```
  FormKe4.Free;
```

```
end;
```

```
procedure TFormKe4.Button1Click(Sender: TObject);  
begin  
  Close;  
end;
```

```
procedure TFormKe4.ComboBox1Change(Sender: TObject);  
var Str,Str1,
```

```
    Str2,Str3,Str4 : String;
```

```
begin
```

```
  Str:= '%s-%s-%s-%s';
```

```
  RadioGroup1.Visible:= True;
```

```
  case ComboBox1.ItemIndex of
```

```
    0 : begin
```

```
      RadioGroup1.Items.Clear;
```

```
      RadioGroup1.Items.Add('<= 1.5/kedalaman sudut');
```

```
      RadioGroup1.Items.Add('> 1.5/kedalaman sudut');
```

```
    end;
```

```
    1 : begin
```

```
      RadioGroup1.Items.Clear;
```

```
      RadioGroup1.Items.Add('<= 1/1000');
```

```
      RadioGroup1.Items.Add('> 1/1000');
```

```
    end;
```

```
    2 : begin
```

```
      RadioGroup1.Items.Clear;
```

```
      RadioGroup1.Items.Add('<= 2.00 mm');
```

```
      RadioGroup1.Items.Add('> 2.00 mm');
```

```
    end;
```

```
    3 : begin
```

```
      RadioGroup1.Items.Clear;
```

```
      RadioGroup1.Items.Add('<= 3.00 mm');
```

```
      RadioGroup1.Items.Add('> 3.00 mm');
```

```
    end;
```

```
  4 : begin
```

```
    RadioGroup1.Items.Clear;
```

```
    RadioGroup1.Items.Add('<= 1.50 mm');
```

```
    RadioGroup1.Items.Add('> 1.50 mm');
```

```
  end;
```

```
  5 : begin
```

```
    RadioGroup1.Items.Clear;
```

```
    RadioGroup1.Items.Add('<= 1.50 mm');
```

```
    RadioGroup1.Items.Add('> 1.50 mm');
```

```
  end;
```

```
end; { of case }
```

```
RadioGroup1.ItemIndex:= 0;
```

```
Str1:= 'FAB';
```

```
Str2:= '04';
```

```
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
```

```
Str4:= IntToStr(RadioGroup1.ItemIndex);
```

```
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
```

```
Edit1.Text:= SearchCode;
```

```
end;
```

```
procedure TFormKe4.ComboBox1Exit(Sender: TObject);
```

```
var Str,Str1,
```

```
    Str2,Str3,Str4 : String;
```

```
begin
```

```
  RadioGroup1.ItemIndex:= 0;
```

```
  Str:= '%s-%s-%s-%s';
```

```
  Str1:= 'FAB';
```

```
  Str2:= '04';
```

```
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
```

```
  Str4:= IntToStr(RadioGroup1.ItemIndex);
```

```
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
```

```
  Edit1.Text:= SearchCode;
```

```
end;
```

```
procedure TFormKe4.RadioGroup1Exit(Sender: TObject);
```

```
var Str,Str1,
```

```
    Str2,Str3,Str4 : String;
```

```
begin
```

```
  Str:= '%s-%s-%s-%s';
```

```
  Str1:= 'FAB';
```

```
  Str2:= '04';
```

```
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
```

```
  Str4:= IntToStr(RadioGroup1.ItemIndex);
```

```
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
```

```
  Edit1.Text:= SearchCode;
```

```
end;
```

```
procedure TFormKe4.RadioGroup1Click(Sender: TObject);
```

```
var Str,Str1,
```

```
    Str2,Str3,Str4 : String;
```

```
begin
```

```
  Str:= '%s-%s-%s-%s';
```

```
  Str1:= 'FAB';
```

```
  Str2:= '04';
```

```
  Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
```

```
  Str4:= IntToStr(RadioGroup1.ItemIndex);
```

```
  SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
```

```
  Edit1.Text:= SearchCode;
```

```
end;
```

```
procedure TFormKe4.Button2Click(Sender: TObject);
```

```
label 100 ;
```

```
begin
```

```
  Checking := 1;
```

```
  FormJunk:= TFormJunk.Create(Self);
```



```

with FormJunk do
begin
try
Table1.DatabaseName:= NamaDrive;
Table1.TableName:='T'fixer.db';
DataSource1.DataSet:= Table1;
DBMemo1.DataSource:= DataSource1;
Table1.Active:= True;
except
on EDatabaseError do
begin
Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
Table1.Active:= False;
Checking:= 0;
FormJunk.Free ;
Exit;
end;
end;

if Checking <> 0 then
begin
Table1.IndexName:='';
if(not Table1.FindKey([SearchCode])) then
begin
Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
FormJunk.Free;
Exit;
end
else
begin
FormJunk.ShowModal;
FormJunk.Free;
end;
end;
end;
end;
end.

```

#### unit AngDB;

interface

uses  
Windows, Messages, SysUtils, Classes, Graphics, Controls,  
Forms, Dialogs,  
StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;

type

```

TFormKe3 = class(TForm)
Bevel1: TBevel;
Label1: TLabel;
ComboBox1: TComboBox;
Bevel2: TBevel;
RadioGroup1: TRadioGroup;
Bevel5: TBevel;
Button1: TButton;
Label3: TLabel;
Bevel6: TBevel;
Edit1: TEdit;
Bevel3: TBevel;
Button2: TButton;
procedure Button1Click(Sender: TObject);
procedure ComboBox1Change(Sender: TObject);
procedure ComboBox1Exit(Sender: TObject);
procedure RadioGroup1Exit(Sender: TObject);
procedure RadioGroup1Click(Sender: TObject);

```

```

procedure Button2Click(Sender: TObject);
private
{ Private declarations }
public
{ Public declarations }
end;

```

procedure Periksa3(NamaDir:String); export;

implementation

uses TDFixer;  
{ \$R \*.DFM }

var

```

FormKe3: TFormKe3;
SearchCode,NamaDrive : String;
Checking : integer;

```

procedure Periksa3(NamaDir:String); export;

begin

```

NamaDrive:= NamaDir;
FormKe3:= TFormKe3.Create(nil);
with FormKe3 do
begin
ComboBox1.Items.Add('Lokasi tepi pelat');
ComboBox1.Items.Add('Permukaan lengkung');
ComboBox1.Items.Add('Lokasi garis cek ; melintang');
ComboBox1.Items.Add('Lokasi garis cek ; memanjang');
ComboBox1.Items.Add('Penyimpangan bentuk ; mal seksi');
ComboBox1.Items.Add('Penyimpangan bentuk pada mal lain');
ComboBox1.ItemIndex:= 0;
RadioGroup1.Visible:= False;
RadioGroup1.ItemIndex:= 0;
Edit1.Text:= '';
end;

```

FormKe3.ShowModal;

FormKe3.Free;

end;

procedure TFormKe3.Button1Click(Sender: TObject);

begin

Close;

end;

procedure TFormKe3.ComboBox1Change(Sender: TObject);

var Str,Str1,

Str2,Str3,Str4 : String;

begin

```

Str:= '%s-%s-%s-%s';
RadioGroup1.Visible:= True;
case ComboBox1.ItemIndex of
0 : begin
RadioGroup1.Items.Clear;
RadioGroup1.Items.Add('<= 2.00 mm');
RadioGroup1.Items.Add('> 2.00 mm');
end;

```

1 : begin

```

RadioGroup1.Items.Clear;
RadioGroup1.Items.Add('<= 2.00 mm');
RadioGroup1.Items.Add('> 2.00 mm');
end;

```



```

2 : begin
    RadioGroup1.Items.Clear;
    RadioGroup1.Items.Add('<= 1.50 mm');
    RadioGroup1.Items.Add('> 1.50 mm');
end;

3 : begin
    RadioGroup1.Items.Clear;
    RadioGroup1.Items.Add('<= 1.50 mm');
    RadioGroup1.Items.Add('> 1.50 mm');
end;

4 : begin
    RadioGroup1.Items.Clear;
    RadioGroup1.Items.Add('<= 1.50 mm');
    RadioGroup1.Items.Add('> 1.50 mm');
end;

5 : begin
    RadioGroup1.Items.Clear;
    RadioGroup1.Items.Add('<= 1.50 mm');
    RadioGroup1.Items.Add('> 1.50 mm');
end;

end; { of case }
RadioGroup1.ItemIndex:= 0;
Str1:= 'FAB';
Str2:= '03';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

```

```

procedure TFormKe3.ComboBox1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    RadioGroup1.ItemIndex:= 0;
    Str:= '%s-%s-%s-%s';
    Str1:= 'FAB';
    Str2:= '03';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

```

```

procedure TFormKe3.RadioGroup1Exit(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'FAB';
    Str2:= '03';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
    Edit1.Text:= SearchCode;
end;

```

```

procedure TFormKe3.RadioGroup1Click(Sender: TObject);
var Str,Str1,
    Str2,Str3,Str4 : String;
begin
    Str:= '%s-%s-%s-%s';
    Str1:= 'FAB';
    Str2:= '03';
    Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
    Str4:= IntToStr(RadioGroup1.ItemIndex);
    SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);

```

```

    Edit1.Text:= SearchCode;
end;

```

```

procedure TFormKe3.Button2Click(Sender: TObject);
label 100 ;
begin
    Checking := 1;
    FormJunk:= TFormJunk.Create(Self);
    with FormJunk do
    begin
        try
            Table1.DatabaseName:= NamaDrive;
            Table1.TableName:= 'Fixer.db';
            DataSource1.DataSet:= Table1;
            DBMemo1.DataSource:= DataSource1;
            Table1.Active:= True;
        except
            on EDatabaseError do
            begin
                Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
                Table1.Active:= False;
                Checking:= 0;
                FormJunk.Free ;
                Exit;
            end;
        end;
    end;
end;

```

```

    if Checking <> 0 then
    begin
        Table1.IndexName:= '';
        if(not Table1.FindKey([SearchCode])) then
        begin
            Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
            FormJunk.Free;
            Exit;
        end
    else
    begin
        FormJunk.ShowModal;
        FormJunk.Free;
    end;
    end;
end;

```

```

end;

```

```

end.
unit HeatDB;

```

```

interface

```

```

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls,
    Forms, Dialogs,
    StdCtrls, ExtCtrls, DBCtrls, Db, DBTables, Mask;

```

```

type

```

```

TFormKe6 = class(TForm)
    Bevel1: TBevel;
    Label1: TLabel;
    ComboBox1: TComboBox;
    Bevel2: TBevel;
    RadioGroup1: TRadioGroup;
    Bevel5: TBevel;
    Button1: TButton;
    Label3: TLabel;
    Bevel6: TBevel;
    Edit1: TEdit;
    Bevel3: TBevel;

```

```

Button2: TButton;
procedure Button1Click(Sender: TObject);
procedure ComboBox1Change(Sender: TObject);
procedure ComboBox1Exit(Sender: TObject);
procedure RadioGroup1Exit(Sender: TObject);
procedure RadioGroup1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
private
{ Private declarations }
public
{ Public declarations }
end;

procedure Periksa6>NamaDir:String); export;

implementation

uses TDFixer;
{$R *.DFM}

var
FormKe6: TFormKe6;
SearchCode>NamaDrive : String;
Checking : integer;

procedure Periksa6>NamaDir:String); export;
begin
>NamaDrive:=>NamaDir;
FormKe6:= TFormKe6.Create(nil);
with FormKe6 do
begin
ComboBox1.Items.Add('Pendinginan dg. air setelah pemanasan');
ComboBox1.Items.Add('Pendinginan dg. udara setelah pemanasan');
ComboBox1.Items.Add('Pendinginan deg air setelah pendinginan dg. udara');
ComboBox1.ItemIndex:=0;
RadioGroup1.Visible:=False;
RadioGroup1.ItemIndex:=0;
Edit1.Text:= '';
end;

FormKe6.ShowModal;

FormKe6.Free;

end;

procedure TFormKe6.Button1Click(Sender: TObject);
begin
Close;
end;

procedure TFormKe6.ComboBox1Change(Sender: TObject);
var Str,Str1,
Str2,Str3,Str4 : String;
begin
Str:= '%s-%s-%s-%s';
RadioGroup1.Visible:= True;
case ComboBox1.ItemIndex of
0 : begin
RadioGroup1.Items.Clear;
RadioGroup1.Items.Add('<= 650 Deg.C');
RadioGroup1.Items.Add('> 650 Deg.C');
end;
1 : begin
RadioGroup1.Items.Clear;
RadioGroup1.Items.Add('<= 900 Deg.C');
RadioGroup1.Items.Add('> 900 Deg.C');

```

```

end;
2 : begin
RadioGroup1.Items.Clear;
RadioGroup1.Items.Add('<= 900 Deg.C');
RadioGroup1.Items.Add('> 900 Deg.C');
end;

end;{ of case }

RadioGroup1.ItemIndex:=0;
Str1:= 'FAB';
Str2:= '06';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe6.ComboBox1Exit(Sender: TObject);
var Str,Str1,
Str2,Str3,Str4 : String;
begin
RadioGroup1.ItemIndex:=0;
Str:= '%s-%s-%s-%s';
Str1:= 'FAB';
Str2:= '06';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe6.RadioGroup1Exit(Sender: TObject);
var Str,Str1,
Str2,Str3,Str4 : String;
begin
Str:= '%s-%s-%s-%s';
Str1:= 'FAB';
Str2:= '06';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe6.RadioGroup1Click(Sender: TObject);
var Str,Str1,
Str2,Str3,Str4 : String;
begin
Str:= '%s-%s-%s-%s';
Str1:= 'FAB';
Str2:= '06';
Str3:= '0'+IntToStr(ComboBox1.ItemIndex+1);
Str4:= IntToStr(RadioGroup1.ItemIndex);
SearchCode:= Format(Str,[Str1,Str2,Str3,Str4]);
Edit1.Text:= SearchCode;
end;

procedure TFormKe6.Button2Click(Sender: TObject);
label 100 ;
begin
Checking := 1;
FormJunk:= TFormJunk.Create(Self);
with FormJunk do
begin
try
Table1.DatabaseName:=>NamaDrive;
Table1.TableName:= 'Fixer.db';
DataSource1.DataSet:= Table1;
DBMemo1.DataSource:= DataSource1;

```



```

    Table1.Active:= True;
except
on EDatabaseError do
begin
    Application.MessageBox('Direktori salah','Error',MB_OK
or MB_ICONERROR);
    Table1.Active:= False;
    Checking:= 0;
    FormJunk.Free;
    Exit;
end;
end;

if Checking <> 0 then
begin
    Table1.IndexName:="";
    if(not Table1.FindKey([SearchCode])) then
    begin
        Application.MessageBox('No matches data
found','Information',MB_OK or MB_ICONINFORMATION);
        FormJunk.Free;
        Exit;
    end
    else
    begin
        FormJunk.ShowModal;
        FormJunk.Free;
    end;
end;
end;

end;

unit inti_8;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls,
    Forms, Dialogs,
    StdCtrls, FileCtrl;

type
    TFormDrive = class(TForm)
        DriveComboBox1: TDriveComboBox;
        DirectoryListBox1: TDirectoryListBox;
        FileListBox1: TFileListBox;
        Button1: TButton;
        Label1: TLabel;
        Label2: TLabel;
        Label3: TLabel;
        FilterComboBox1: TFilterComboBox;
        Edit1: TEdit;
        Label4: TLabel;
        Label5: TLabel;
        Button2: TButton;
        procedure FormCreate(Sender: TObject);
        procedure Button1Click(Sender: TObject);
        procedure Button2Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    FormDrive : TFormDrive;
    ActDir,CurrentDir,
    DefaultDir : String;
    CheckState : integer;

```

implementation

```

{$R *.DFM}

procedure TFormDrive.FormCreate(Sender: TObject);
begin
    FileListBox1.FileEdit:= Edit1;
    DirectoryListBox1.FileList:= FileListBox1;
    DirectoryListBox1.DirLabel:= Label1;
    DriveComboBox1.DirList:= DirectoryListBox1;
    FilterComboBox1.FileList:= FileListBox1;
    FilterComboBox1.Filter:='Database Files|*.db;*.dbf';
end;

procedure TFormDrive.Button1Click(Sender: TObject);
begin
    CurrentDir:= DirectoryListBox1.Directory;
    CheckState := 1;
    Close;
end;

procedure TFormDrive.Button2Click(Sender: TObject);
begin
    ActDir:= CurrentDir;
    Close;
end;

end.

unit Erection;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls,
    Forms, Dialogs,
    StdCtrls, ExtCtrls;

type
    TFormErection = class(TForm)
        Bevel1: TBevel;
        RadioGroup1: TRadioGroup;
        Bevel2: TBevel;
        Bevel3: TBevel;
        Button1: TButton;
        Button2: TButton;
        procedure Button1Click(Sender: TObject);
        procedure Button2Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

procedure ErectionCek>NamaDir : String);export;

implementation

{$R *.DFM}
uses inti_8;

var
    FormErection : TFormErection;
   >NamaDrive : String;

procedure CallingDLL>NamaFile,>NamaProc,>NamaKendali :
String);
type
    TProTestDLL = procedure>NamaSopir:String);

```



```

var
  ProTestDLL : TProTestDLL;
  ABuffer, Buffer1 : array[0..255] of Char;
  libHandle : THandle;
begin
  StrPCopy(Buffer1, NamaFile);
  libHandle := LoadLibrary(Buffer1);
  if (libHandle = 0) then
    begin
      MessageDlg('Can not find the DLLs', mtError, [mbOk], 0);
      exit;
    end;

  StrPCopy(ABuffer, NamaProc);

  @ProTestDLL := GetProcAddress(libHandle, ABuffer);
  ProTestDLL(NamaKendali);

  FreeLibrary(libHandle);
end;

procedure ErectionCek(NamaDir : String); export;
begin
  NamaDrive := NamaDir;
  FormErection := TFormErection.Create(nil);
  with FormErection do
    begin
      RadioGroup1.Items.Clear;
      RadioGroup1.Items.Add('Mis-alignment');
      RadioGroup1.Items.Add('Celah/Gap');
      RadioGroup1.ItemIndex := 0;
    end;
  FormErection.ShowModal;
  FormErection.Free;
end;

procedure TFormErection.Button1Click(Sender: TObject);
begin
  Close;
end;

procedure TFormErection.Button2Click(Sender: TObject);
begin
  case RadioGroup1.ItemIndex of
    0 : CallingDLL('ERECT_DB.DLL', 'Periksa 14', NamaDrive);
    1 : CallingDLL('ERECT_DB.DLL', 'Periksa 15', NamaDrive);
  end;
end;

end.

```

#### library REPAIRS;

```

uses
  Erection in 'Erection.pas' {FormErection},
  Fabrik in 'Fabrik.pas' {FormFabrikasi},
  SubAss in 'SubAss.pas' {FormSubAssembly},
  AssMas in 'AssMas.pas' {FormAssembly};

{$R *.RES}
exports FabrikasiCek index 1;
exports SubAssyCek index 2;
exports AssemblyCek index 3;
exports ErectionCek index 4;
begin
end.

```

#### unit CetakMas;

##### interface

##### uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs, StdCtrls, ExtCtrls, DBCtrls, Mask, Db, DBTables, Buttons;

##### type

```

TFormCetakData = class(TForm)
  Bevel1: TBevel;
  Bevel3: TBevel;
  Label5: TLabel;
  Label1: TLabel;
  Label3: TLabel;
  Label7: TLabel;
  Label9: TLabel;
  ComboBox1: TComboBox;
  Bevel4: TBevel;
  Edit1: TEdit;
  Edit3: TEdit;
  Edit5: TEdit;
  Edit7: TEdit;
  BitBtn1: TBitBtn;
  Bevel2: TBevel;
  BitBtn2: TBitBtn;
  procedure FormCreate(Sender: TObject);
  procedure BitBtn1Click(Sender: TObject);
  procedure Edit1Exit(Sender: TObject);
  procedure Edit3Exit(Sender: TObject);
  procedure Edit5Exit(Sender: TObject);
  procedure ComboBox1Change(Sender: TObject);
  procedure ComboBox1Exit(Sender: TObject);
  procedure BitBtn2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }

```

##### protected

```

procedure EvGetMinMaxInfo(var M: TWMGetMinMaxInfo);
  message WM_GETMINMAXINFO;

end;

```

##### var

```

FormCetakData : TFormCetakData;
CodeCari, Str1,
Str2, Str3, Str4, Str : String;
const
  eKeyViol = 9729;
  eRequiredFieldMissing = 9732;
  Lebar = 358;
  Tinggi = 280;

```

##### implementation

```
{ $R *.DFM }
```

```
uses RepKontes, Inti_8;
```

```

procedure TFormCetakData.EvGetMinMaxInfo(var M:
TWMGetMinMaxInfo);
begin
  inherited;

```

```

  M.MinMaxInfo^.ptMinTrackSize.x := lebar;
  M.MinMaxInfo^.ptMaxTrackSize.x := lebar;
  M.MinMaxInfo^.ptMinTrackSize.y := tinggi;
  M.MinMaxInfo^.ptMaxTrackSize.y := tinggi;

```

```

M.MinMaxInfo^.ptMaxSize.x:= lebar+5;
M.MinMaxInfo^.ptMaxSize.y:= tinggi+5;
M.MinMaxInfo^.ptMaxPosition.x:= 20;
M.MinMaxInfo^.ptMaxPosition.y:= 20;

end;

procedure TFormCetakData.FormCreate(Sender: TObject);
begin
  ComboBox1.Items.Add('Dibawah standar');
  ComboBox1.Items.Add('Diatas standar');
  ComboBox1.ItemIndex:= 0;
  Edit1.Text:= 'FAB';
  Edit3.Text:= '01';
  Edit5.Text:= '01';
  Edit7.Text:= ' ';
end;

procedure TFormCetakData.BitBtn1Click(Sender: TObject);
begin
  CheckState:= 1;
  FormRepKontes:= TFormRepKontes.Create(Self);
  with FormRepKontes do
    begin
      try
        Table1.DatabaseName:= ActDir;
        Table1.TableName:= 'Baikin.DB';
        Table1.Active:= True;
      except
        on EDatabaseError do
          begin
            Application.MessageBox('Direktori
salah....', 'Seweng', MB_OK or MB_ICONWARNING);
            Table1.Active:= False;
            CheckState:= 0;
            FormRepKontes.Free;
            Exit;
          end;
        end;
      if CheckState <> 0 then
        begin
          Table1.IndexName:= '';
          if (not Table1.FindKey([CodeCari])) then
            begin
              Application.MessageBox('Data tidak
ditemukan', 'Information', MB_OK or
MB_ICONINFORMATION);
              FormRepKontes.Free;
              Exit;
            end
          else
            begin
              BaikRep.DataSet:= Table1;
              Table1.SetRange([CodeCari], [CodeCari]);
              BaikRep.Preview;
              BaikRep.Page.PaperSize:=
BaikRep.PrinterSettings.PaperSize;
              Table1.CancelRange;
              Table1.Active:= False;
              FormRepKontes.Free;
            end;
          end;
        end;
      end;
    end;

procedure TFormCetakData.Edit1Exit(Sender: TObject);
begin
  Str:= '%s-%s-%s-%s';
  Str1:= Edit1.Text;
  Str2:= Edit3.Text;

```

```

  Str3:= Edit5.Text;
  Str4:= IntToStr(ComboBox1.ItemIndex);
  CodeCari:= Format(Str, [Str1, Str2, Str3, Str4]);
  Edit7.Text:= CodeCari;
end;

procedure TFormCetakData.Edit3Exit(Sender: TObject);
begin
  Str:= '%s-%s-%s-%s';
  Str1:= Edit1.Text;
  Str2:= Edit3.Text;
  Str3:= Edit5.Text;
  Str4:= IntToStr(ComboBox1.ItemIndex);
  CodeCari:= Format(Str, [Str1, Str2, Str3, Str4]);
  Edit7.Text:= CodeCari;
end;

procedure TFormCetakData.Edit5Exit(Sender: TObject);
begin
  Str:= '%s-%s-%s-%s';
  Str1:= Edit1.Text;
  Str2:= Edit3.Text;
  Str3:= Edit5.Text;
  Str4:= IntToStr(ComboBox1.ItemIndex);
  CodeCari:= Format(Str, [Str1, Str2, Str3, Str4]);
  Edit7.Text:= CodeCari;
end;

procedure TFormCetakData.ComboBox1Change(Sender:
TObject);
begin
  Str:= '%s-%s-%s-%s';
  Str1:= Edit1.Text;
  Str2:= Edit3.Text;
  Str3:= Edit5.Text;
  Str4:= IntToStr(ComboBox1.ItemIndex);
  CodeCari:= Format(Str, [Str1, Str2, Str3, Str4]);
  Edit7.Text:= CodeCari;
end;

procedure TFormCetakData.ComboBox1Exit(Sender: TObject);
begin
  Str:= '%s-%s-%s-%s';
  Str1:= Edit1.Text;
  Str2:= Edit3.Text;
  Str3:= Edit5.Text;
  Str4:= IntToStr(ComboBox1.ItemIndex);
  CodeCari:= Format(Str, [Str1, Str2, Str3, Str4]);
  Edit7.Text:= CodeCari;
end;

procedure TFormCetakData.BitBtn2Click(Sender: TObject);
begin
  Close;
end;

end.

unit Suwun;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls,
  Forms, Dialogs,
  StdCtrls, ExtCtrls, Buttons;

type
  TFormSuwun = class(TForm)
    Bevel1: TBevel;

```



```

Bevel2: TBevel;
Button1: TButton;
BitBtn1: TBitBtn;
procedure Button1Click(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure FormDestroy(Sender: TObject);
procedure FormPaint(Sender: TObject);
private
{ Private declarations }
isAnimated : Boolean;
bmpText : TBitmap;
Posisi : integer;
Box : TRect;
TotHeight : integer;
StrList : TStringList;

procedure InitText;
public
{ Public declarations }
procedure EvTimer(var M: TMessage);
message WM_TIMER;
end;

const
ID_TIMER = 101;

var
FormSuwun: TFormSuwun;

implementation

{$R *.DFM}

procedure TFormSuwun.Button1Click(Sender: TObject);
begin
Close;
end;

procedure TFormSuwun.BitBtn1Click(Sender: TObject);
var Tinggi, i : integer;
begin
if IsAnimated then
begin
KillTimer(Handle, ID_TIMER);

bmpText.Free;
StrList.Free;

IsAnimated:= False;

BitBtn1.Font.Name:= 'Times New Roman';
BitBtn1.Font.Style:= [fsBold];
BitBtn1.Font.Size:= 12;
BitBtn1.Font.Color:= clRed;
BitBtn1.Caption:= 'Terima kasih';
end
else
begin
InitText;

IsAnimated:= True;
Tinggi:= Canvas.TextHeight('A');
Posisi:= Box.Bottom;
TotHeight:= StrList.Count*Tinggi;

BitBtn1.Font.Name:= 'Times New Roman';
BitBtn1.Font.Style:= [];
BitBtn1.Font.Size:= 12;

```

```

BitBtn1.Font.Color:= clRed;
BitBtn1.Caption:= '&Stop';

bmpText:= TBitmap.Create;
bmpText.Width:= Box.Right-Box.Left;
bmpText.Height:= TotHeight;
bmpText.Canvas.Font.Assign(Canvas.Font);
SetBkMode(bmpText.Canvas.Handle, TRANSPARENT);
for i:= 0 to (StrList.Count-1) do
bmpText.Canvas.TextOut(0, i*Tinggi, StrList.Strings[i]);

InvalidateRect(Handle, @Box, False);

SetTimer(Handle, ID_TIMER, 50, nil);
end;
end;

procedure TFormSuwun.InitText;
begin
StrList:= TStringList.Create;
with StrList do
begin
Add(' Ucapan terima kasih kepada : ');
Add(' - Ayahanda tercinta.....');
Add(' - Ibunda tersayang....');
Add(' untuk beliau berdua ..');
Add(' Thank you for teaching me...');
Add(' how to survive in this fuckin world');
Add(' - All my sisters and my brothers');
Add(' ');
Add(' ');
Add(' especially to the beloved XXXX');
Add(' ');
Add(' ');
Add(' - Mr. Ir. Sjarief Widjaja, MSe, Ph.D');
Add(' in capability as our Advisory Lecture');
Add(' - Mr. Ir. Soejitno, - FTK - ITS. ');
Add(' in capability as our Advisory Lecture');
Add(' - VHL Staffs');
Add(' ');
Add(' V-Ins 2.4 would not ');
Add(' have been what it is today');
Add(' without the help and support');
Add(' from all theese people');
Add(' ');
Add(' ');
Add(' At last....');
Add(' we just can say...');
Add(' what is understood.....');
Add(' .....need not to be discussed');

end;
end;

procedure TFormSuwun.EvTimer(var M: TMessage);
begin
if (Posisi > Box.Top) then
BitBlt(Canvas.Handle, Box.Left, Posisi, Box.Right-Box.Left,
Box.Bottom-
Posisi, bmpText.Canvas.Handle, 0, 0, SRCCOPY)
else
BitBlt(Canvas.Handle, Box.Left, Box.Top, Box.Right-Box.Left,
Box.Bottom-Box.Top, bmpText.Canvas.Handle, 0, Box.Top-
Posisi, SRCCOPY);

if (Posisi+TotHeight < Box.Top) then Posisi:= Box.Bottom
else dec(Posisi);
end;

procedure TFormSuwun.FormCreate(Sender: TObject);

```



```

begin
  Box:= Rect(20,20,460 ,165);
  Canvas.Font.Name:='BankGothic Md B1';
  Canvas.Font.Style:=[];
  Canvas.Font.Size:= 12;
  Canvas.Font.Color:=clActiveCaption;
end;

procedure TFormSuwun.FormDestroy(Sender: TObject);
begin
  if IsAnimated then
    begin
      KillTimer(Handle,ID_TIMER);
      bmpText.Free;
      StrList.Free;
    end;
end;

procedure TFormSuwun.FormPaint(Sender: TObject);
begin
  Canvas.Rectangle(Box.Left-1,Box.Top-
1,Box.Right+1,Box.Bottom+1);
end;

end.

```

# Konsep dan Perancangan Program Database

## PENDAHULUAN

Ada beberapa aplikasi perancangan program database yang sekarang ini banyak ditemui, diantaranya adalah : *dBASE III, dBASE IV, FoxBase, FoxPRO, Paradox, Clipper* dan lain-lain. Tetapi masing-masing aplikasi program tersebut mempunyai kelemahan masing-masing sehingga seorang *programmer* harus dapat memilih jenis aplikasi program yang sesuai agar penerapannya akan memudahkan si pemakai.

Teknik perancangan program database ada dua metode yang digunakan yaitu :

metode Normalisasi

metode *Entity Relationship*

Masing-masing metode diatas saling melengkapi satu sama lain. Pilihan untuk metode yang akan digunakan tergantung oleh *programmer* itu sendiri. Untuk tingkat pemula sebaiknya menggunakan metode Normalisasi. Sedangkan untuk yang sudah berpengalaman sebaiknya memakai *Entity Relationship*.

## DEFINISI

Kita mengenal beberapa definisi yang akan digunakan di dalam perancangan program *database* yaitu :

**ENTITY :**

*Entity* adalah orang, tempat, kejadian atau konsep yang informasinya direkam.

#### ATTRIBUTE FIELD :

Setiap *entity* mempunyai *atribute* (*field*) atau seutan untuk mewakili suatu *entity*.

#### DATA VALUE :

Yaitu suatu data aktual atau informasi yang disimpan pada tiap *atribute* atau *field*.

#### RECORD / TUPLE :

*Record* atau *tuple* adalah kumpulan elemen *atribute* beserta data *valuenya* yang saling berkaitan dalam menginformasikan *entity* secara lengkap. Satu *record* / *tuple* mewakili satu data atau informasi tentang sesuatu *entity*.

#### FILE :

*File* adalah kumpulan *record-record* sejenis yang mempunyai panjang elemen yang sama, *atribute* yang sama, namun berbeda-beda data *value* tiap-tiap *atributenya*.

#### DATABASE :

*Database* adalah kumpulan *file-file* yang saling berkaitan antara satu *file* dengan *file* yang lain sehingga membentuk satu bangunan data untuk menginformasikan satu perusahaan, instansi dalam batasan tertentu. Jika terdapat *file* yang tidak dapat dipadukan atau dihubungkan dengan *file* yang lainnya, berarti *file* tersebut bukanlah kelompok dari satu *database*. Ia akan membentuk satu *database* sendiri.

### KONSEP DATABASE MANAGEMENT SISTEM

*Database Management Sistem* ( DBMS ) yaitu sekumpulan data yang



saling berhubungan dan satu set program untuk mengakses data tersebut.

*Database Management Sistem* terdiri dari :

## DATABASE

*Database* adalah kumpulan *file-file* yang saling berelasi, dimana relasi tersebut biasanya ditunjukkan dengan kunci dari tiap *file* yang ada. Satu *database* menunjukkan satu kumpulan data yang dipakai dalam satu lingkup perusahaan, instansi. Dalam satu *file* terdapat *record-record* yang sejenis, sama besar, sama bentuk, dan merupakan kumpulan *entity* yang seragam. Satu *record* terdiri dari *field-field* yang saling berhubungan untuk menunjukkan bahwa *field* tersebut dalam satu pengertian yang lengkap dan direkam dalam satu *record*. Apabila kita ingin menyebut isi dari *field* maka digunakan *atribute* atau merupakan judul dari suatu kelompok *entity* tertentu.

## PROGRAM PENGELOLA

Program pengelola merupakan satu paket program yang dibuat agar memudahkan dan mengefisienkan pengorganisasian data ke dalam *database*, seperti :

- menambah data
- menghapus data
- mengambil data
- membaca data
- membuat *query*
- mengurutkan data
- mencetak data

Hubungan antara definisi di atas dan antara *database* dengan program

pengelolanya dapat dilihat pada gambar dibawah ini :

## GAMBARAN DBMS

### PERANAN *DATABASE* PROGRAM

Didalam suatu perusahaan sering didapati berbagai permasalahan yang timbul akibat dari rumitnya manajemen data yang ada. Masalah-masalah di atas dapat dikelompokkan menjadi beberapa diantaranya :

- *Redundansi* dan *Inkonsistensi* data
- Kesulitan pengaksesan data
- Isolasi data untuk Standarisasi
- *Multiple user* ( banyak pemakai )
- Masalah data *secutity* ( keamanan data )
- Masalah integrasi data ( kesatuan )
- Masalah *independence* data ( kebebasan data )

Masalah-masalah diatas dapat dijelaskan dibawah ini :

### **REDUNDANSI DAN INKONSISTENSI DATA**

Apabila *file-file database* dan program pengelolanya dibuat oleh *programmer* yang berbeda pada waktu yang berselang lama, maka ada beberapa bagian data yang mengalami penggandaan pada *file-file* yang berbeda dalam satu *database*. **Penyimpangan data di beberapa tempat untuk data yang sama disebut redundansi.** Adanya *redundansi* akan mengakibatkan pemborosan pada ruang penyimpanan data. Disamping itu biaya untuk mengakses data menjadi lebih tinggi karena pengerjaan mengakses data harus dilakukan di beberapa *file* yang berbeda.

Penyimpangan data yang berulang-ulang di beberapa *file* dapat



mengakibatkan terjadinya *inkonsistensi* data.

## **KESULITAN DALAM PENGAKSESAN DATA**

Apabila disuatu perusahaan dibutuhkan data-data perhitungan dengan suatu spesifik khusus maka pencetakan akan sulit dilakukan mengingat kriteria yang sesuai dengan yang diminta tidak terdapat dalam program yang dibuat untuk mencetak data tersebut. Penyelesaian masalah diatas adalah kearah DBMS yang mampu mengambil data secara langsung dengan bahas yang *familiar* dan mudah digunakan (*user friendly*) yaitu melalui penyediaan menu-menu program pengelolanya.

## **ISOLASI DATA UNTUK STANDARISASI**

Jika data tersebar dalam beberapa *file* dalam bentuk format yang tidak sama, maka ini akan menyulitkan dalam menulis program pengelolanya untuk mengambil dan menyimpan data. Oleh karena itu data dalam suatu *database* haruslah dibuat satu format yaitu *format text database* sehingga mudah dibuat program pengelolanya. Bisa dibayangkan bagaimana sulitnya membuat program pengelola *database* bila datanya dibuat dari campuran *format text file Pascal, Basic, C++, Lotus.123* dan sebagainya.

## **MULTIPLE USER ( BANYAK PEMAKAI )**

Untuk mempercepat semua daya guna sistem dan mendapat responsi waktu yang cepat, beberapa sistem mengijinkan banyak pemakai untuk meng*update* data secara bersama-sama maupun bergantian. Salah satu alasan mengapa *database* dibangun, karena nantinya data tersebut digunakan oleh banyak orang dalam waktu yang sama atau berbeda, diakses oleh program yang sama tapi berbeda orang dan waktu.

## **MASALAH DATA SECURITY ( KEAMANAN DATA )**



Tidak semua pemakai sistem *database* diperbolehkan untuk mengakses semua data dan melakukan pengorganisasian ( pengaturan dan pengubahan ) data yang telah diaksesnya. Keamanan data ini dapat diatur lewat program yang dibuat oleh *programmer* atau fasilitas keamanan dari *operating sistemnya* misalnya :

*Novell Netware* untuk *Local Area Network*

*Direct Acces* untuk *Personal Computer*

### **MASALAH INTEGRITAS DATA ( KESATUAN DATA )**

*Database* berisi *file-file* yang saling berkaitan, masalah utama adalah bagaimana keterkaitan antara *file* tersebut terjadi. Jadi yang utama adalah *field* kunci apa yang mengkaitkan satu *file* dan *file* lainnya dalam satu *database*.

### **MASALAH DATA INDEPENDENCE ( KEBABASAN DATA )**

Pada suatu aplikasi yang kita buat dengan bahasa pemrograman *BASIC* misalnya, bila program telah kita buat untuk menyelesaikan pembacaan data untuk *file* tertentu maka apabila program telah jadi dan terdapat perubahan struktur pada *file* tersebut maka program tersebut haruslah diubah. Dapat dikatakan apabila struktur datanya berubah maka struktur program pengelolanya juga berubah. Hal ini berarti bahwa program pengelola *database* yang kita buat tidak bebas terhadap *database* yang ada. Berbeda dengan paket bahasa yang menggunakan pengolahan DBMS ( *Database Management System* ), apapun yang terjadi pada struktur *file*, setiap kita akan :  
melihat / mencari data cukuplah dengan perintah *SEARCH* atau *FIND*  
menambah data cukup dengan perintah *APPEND* atau *NEW*  
mencari data sesuai dengan kriteria tertentu cukup dengan *QUERY*

Hal ini berarti perintah-perintah dalam paket DBMS bebas terhadap *database* yang dikelolanya. Apapun perubahan dalam *database*, semua perintah akan mengalami kestabilan tanpa perlu diubah-ubah. Namun perlu pula dipikirkan bagaimana bila ada syarat-syarat terhadap *database* yang ada.

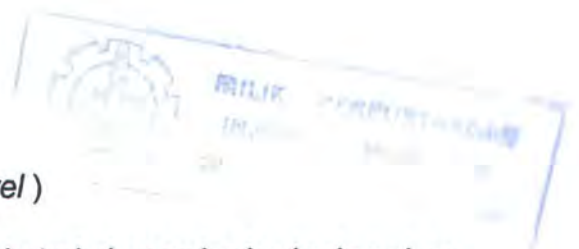
## **ABSTRAKSI DATA ( DATA ABSTRACTION )**

*Database Management System* mempunyai kegunaan agar sipemakai / *user* mampu menyusun suatu pandangan abstraksi dari data. Bayangan mengenai data tidak lagi memperhatikan kondisi sesungguhnya “Bagaimana” suatu data masuk ke dalam *database*, “Bagaimana” proses penyimpanannya dalam *disk* (media penyimpan), di sektor manakah ia disimpan, tetapi menyangkut secara menyeluruh bagaimana data tersebut dapat diabstraksikan / digambarkan menyerupai kondisi yang dihadapi pemakai sehari-hari. Adapun sistim yang sesungguhnya tentang teknis bagaimana data disimpan dan dipelihara seakan disembunyikan kerumitannya dan kemudian diungkapkan dalam bahasa dan gambar yang mudah dimengerti oleh orang awam.

Jadi pada dasarnya **abstraksi data adalah suatu kemampuan data untuk membentuk suatu pandangan tertentu pada si pemakai. Pemakai / user dapat dikelompokkan menjadi 3 tingkatan abstraksi saat memandang suatu *database* yaitu :**

- *Level Fisik*
- *Level Konseptual*
- *Level Pandangan Pemakai ( View Level )*

Masing-masing *level* mempunyai tingkat abstraksi yang berbeda dan akan





dijabarkan sebagai berikut:

### **LEVEL FISIK :**

*Level* fisik adalah *level* abstraksi paling rendah. *Level* ini menggambarkan “bagaimana” data disimpan dalam kondisi yang sebenarnya, seperti :

- Bagaimana proses data diubah dalam bentuk bahasa mesin.
- Bagaimana proses penambahan *sector* bila data telah ditambah dan disimpan dalam media penyimpanan.
- Bagaimana proses pembagian *sector* untuk data yang tidak sama. *Level* ini merupakan *level* paling kompleks. Struktur data *level* terendah terletak dalam *level* ini. *Level* ini digunakan oleh seorang *System Analyst*.

### **LEVEL KONSEPTUAL**

*Level* konseptual merupakan *level* abstraksi data yang lebih tinggi daripada *Level* Fisik. Ia menggambarkan data apa yang disimpan dalam *database*, dan hubungan yang terjadi antar data. *Level* ini keseluruhan *database*. Pemakai tidak memperhatikan kerumitan dalam struktur *Level* Fisik lagi. Penggambaran cukup dengan memakai kontak, garis keterrangan secukupnya. *Level* ini digunakan oleh *database administrator ( Programmer )*, yang akan memutuskan informasi apa saja yang akan dipakai dan dipelihara oleh *user* dalam satu *database*.

### **LEVEL PANDANGAN PEMAKAI ( VIEW LEVEL )**

*Level* pandangan pemakai ini merupakan *level* abstraksi tertinggi yang menggambarkan hanya satu bagian dri keseluruhan data. Bila pada *level* konseptual data merupakan suatu kumpulan informasi yang besar dan kompleks, maka pada *level* ini data hanya sebagian saja yang dilihat dan



dipakai. Hal ini disebabkan oleh karena beberapa pemakai / *user* program *database* yang dibuat tidak membuat semua isi *database*. *Level* ini sangat dekat dengan pemakai / *user*. Setiap *user* butuh sebagian dari *database*. Ada beberapa kelompok *user* dengan pandangan berbeda yang membutuhkan data dalam *database*. Orang yang bekerja pada *level* ini sering disebut sebagai pemakai / *user* program. Jadi tingkatan abstraksi di atas didefinisikan untuk memudahkan hubungan antara pemakai / *user* program pada *level* Pandangan Pemakai dengan program pengelola program *database* yang dibuat pada *Level Konseptual*. Adapun hubungan antara ketiga *level* diatas dapat digambarkan seperti dibawah ini :

Konsep dari *Level Abstraksi Data* ini akan mempermudah pengertian mengenai kebebasan data ( *Data Independence* ). Kebebasan data dapat dibagi menjadi 2 bagian utama yaitu :

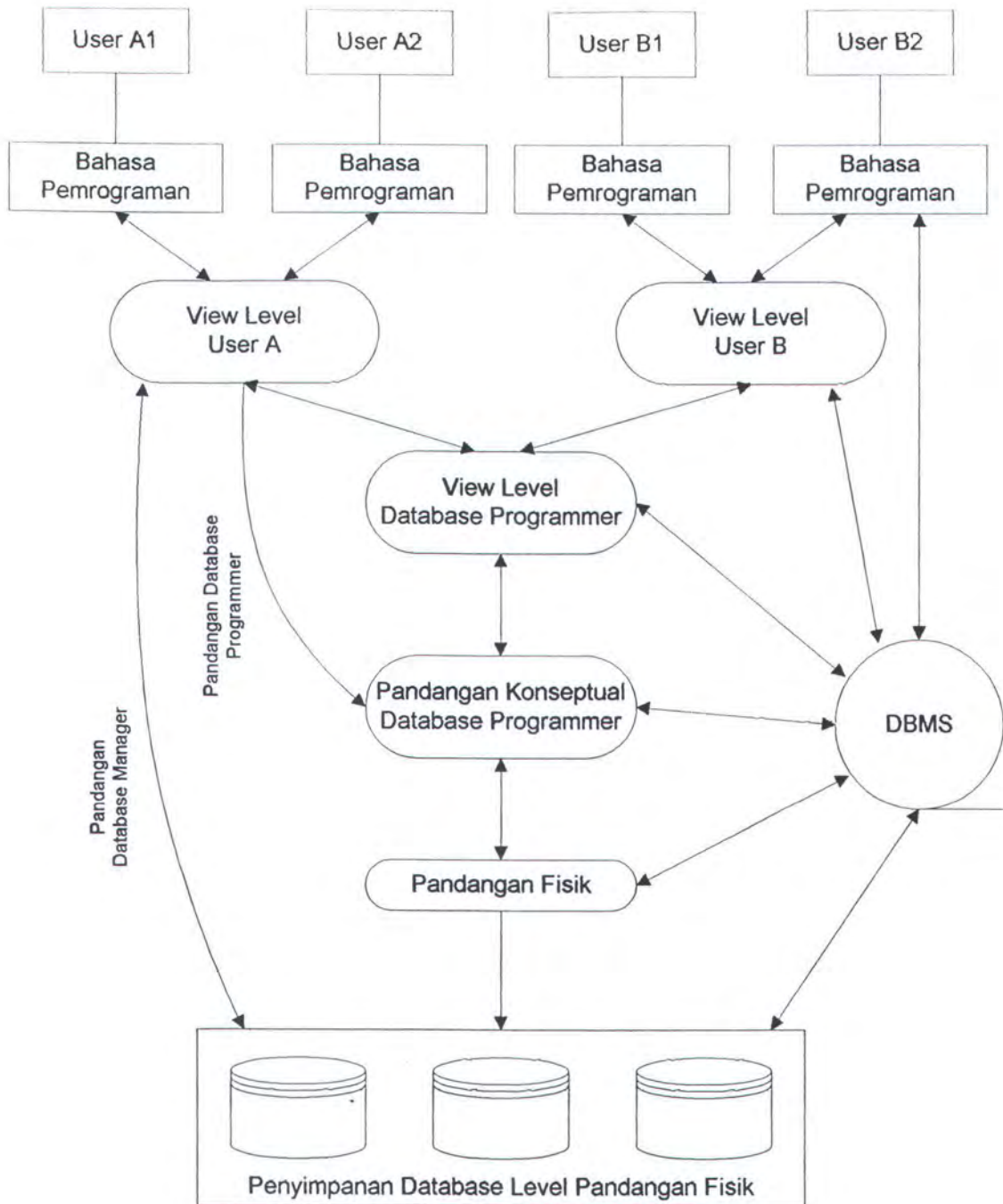
*Phisycal Data Independence* :

Yaitu kebebasan untuk mengubah bentuk tampilan fisik *database* tanpa mengakibatkan suatu aplikasi program ditulis kembali. Modifikasi ini dilakukan pada *level* fisik dan dimaksudkan untuk meningkatkan daya guna DBMS.

*Logical Data Independence* :

Yaitu kebebasan untuk mengubah bentuk konseptual program pengelola *database* sehingga mengakibatkan suatu aplikasi program ditulis kembali. Modifikasi pada *Logical Data Independence* ini berada pada *Level Konseptual* sehingga merubahnya berarti merubah struktur program pengelola *datasenya*.

Dengan menggunakan *Level Abstraksi Data* diatas maka kita dapat lebih mudah memahami *Arsitektur Database Management System* seperti yang kita lihat pada gambar dibawah. Gambar ini memperlihatkan bagaimana peranan seorang *Database Administrator / Programmer* dan apa saja yang ditangani oleh DBMS secara *global*, dipandang dari berbagai *level*.



Gambar Arch. DBMS.

## PAKET BAHASA

Dalam DBMS terdapat paket bahasa yang berguna untuk pengorganisasian *database* yang ada. Di sini ada 3 bahasa yang termasuk dalam bahasa pengelola DBMS yaitu :



- *Data Definition Language ( DDL )*
- *Data Manipulation Language ( DML )*
- *Query Language*

Ketiga bahasa diatas mempunyai kegunaan masing-masing di dalam pengorganisasian, tapi mereka akan bekerja bersama-sama dalam proses pengorganisasian. Masing-masing bahasa diatas akan dijelaskan sebagai berikut :

### ***DATA DEFINITION LANGUAGE ( DDL ) :***

*Data Definition Language* adalah satu bahasa yang berisi satu set definisi perintah khusus untuk menspesifikasikan pola *database* yang diorganisasikan. Hasil spesifikasi dari perintah-perintah khusus tersebut adalah satu set tabel yang disimpan dalam *file* khusus yang disebut *Data Dictionary* atau *Directory*. Satu *Directory* adalah satu *file* yang berisi *metadata*, yaitu “data” mengenai data. *File* ini dikonsultasikan dahulu sebelum data sebenarnya dibaca atau dimodifikasi dalam satu sistem *database*. Struktur penyimpanan dan metode akses yang digunakan sistem *database* dispesifikasikan dengan satu set definisi dalam satu tipe DDL yang disebut *Data Storage* dan *Definition Language*. Hasil kompilasi dari definisi ini adalah satu set perintah yang menspesifikasikan suatu terapan yang rinci dari pola *database* yang biasanya tersembunyi dari *user/* pemakai. Contoh dari perintah ini adalah : *CREATE, MODIFY, SETUP, NEW*.

### ***DATA MANIPULATION LANGUAGE ( DML )***

*Data Manipulation Language ( DML )* adalah bahasa yang mengijinkan pemakai untuk mengakses dan memanipulasi data sebagai yang telah diorganisasikan sebelumnya dalam model data yang tepat.

Jadi DML mengizinkan pemakai / *user* untuk :

- Mengambil informasi baru ke dalam *database*
- Menyisipkan informasi baru ke dalam *database*
- Menghapus informasi dari *database*

Di dalam DML ada 2 tipe yaitu :

- *Procedural DML* :

DML yang membutuhkan masukan dari pemakai untuk menspesifikasikan data apa saja yang dibutuhkan dan bagaimana cara untuk mendapatkannya.

- *Non Procedural DML* :

DML yang membutuhkan masukan dari pemakai untuk menspesifikasikan data apa saja yang dibutuhkan tanpa menspesifikasikan bagaimana cara untuk mendapatkannya. Tipe *Non Procedural DML* ini sangat mudah digunakan dan dipelajari oleh pemakai / *user* dibanding dengan *Procedural DML*. Keduanya akan dipakai dalam pembuatan program *database* Standarisasi Penyimpangan dan Perbaikan pada Proses Produksi Kapal ini.

## **QUERY**

*Query* adalah pernyataan yang berisi kriteria tertentu untuk diajukan dalam mengambil suatu informasi menurut kriteria tersebut. Berdasarkan cara kerjanya. *Query* ini termasuk *DML Language*.

## **PENGGUNA DATABASE**

Dilihat dari otoritasnya terhadap suatu *database*, maka dikenal 3 jenis pengguna yaitu:

- *Database Manager*





- *Database Administrator*
- *Database User*

Ketiga *database* diatas dapat dijelaskan sebagai berikut :

### **DATABASE MANAGER**

*Database Manager* adalah orang yang bertugas menyediakan suatu *modul* program yang berisi tentang hubungan antara penyimpanan data *Level Fisik* dalam *database* dengan suatu aplikasi program dan *query* yang diajukan ke *modul*. Dengan kata lain, *database manager* merupakan orang yang membuat bahasa yang akan dipakai untuk mengolah program *database*.

Misalnya :

- Perusahaan pembuat *Program Dbase* yaitu *Ashon Tate*
- Perusahaan pembuat *Program FoxPRO* yaitu *Microsoft Cooperation*.

### **DATABASE ADMINISTRATOR**

*Database Asministrator* adalah orang yang mempunyai kekuasaan sebagai pembuat pusat pengontrolan terhadap seluruh *database* dan program *database* yang dibuatnya untuk mengolah *database* tersebut. Dengan kata lain *Database Administrator* adalah *programmer database*. Fungsi *Database Administrator* adalah :

- Mendefinisikan pola *struktur database*
- Mendefinisikan struktur penyimpanan dan metode akses program
- Mampu memodifikasi pola program daan organisasi *database*
- Memberikan kekuasaan pada *user* untuk mengakses *database*
- Menspesifikasikan integritas suatu data

### **DATABASE USER**

*Database User* adalah orang yang berhak untuk memakai dan



menjalankan program *database*. Berdasarkan pengalamannya *Database User* dibagi menjadi 2 yaitu :

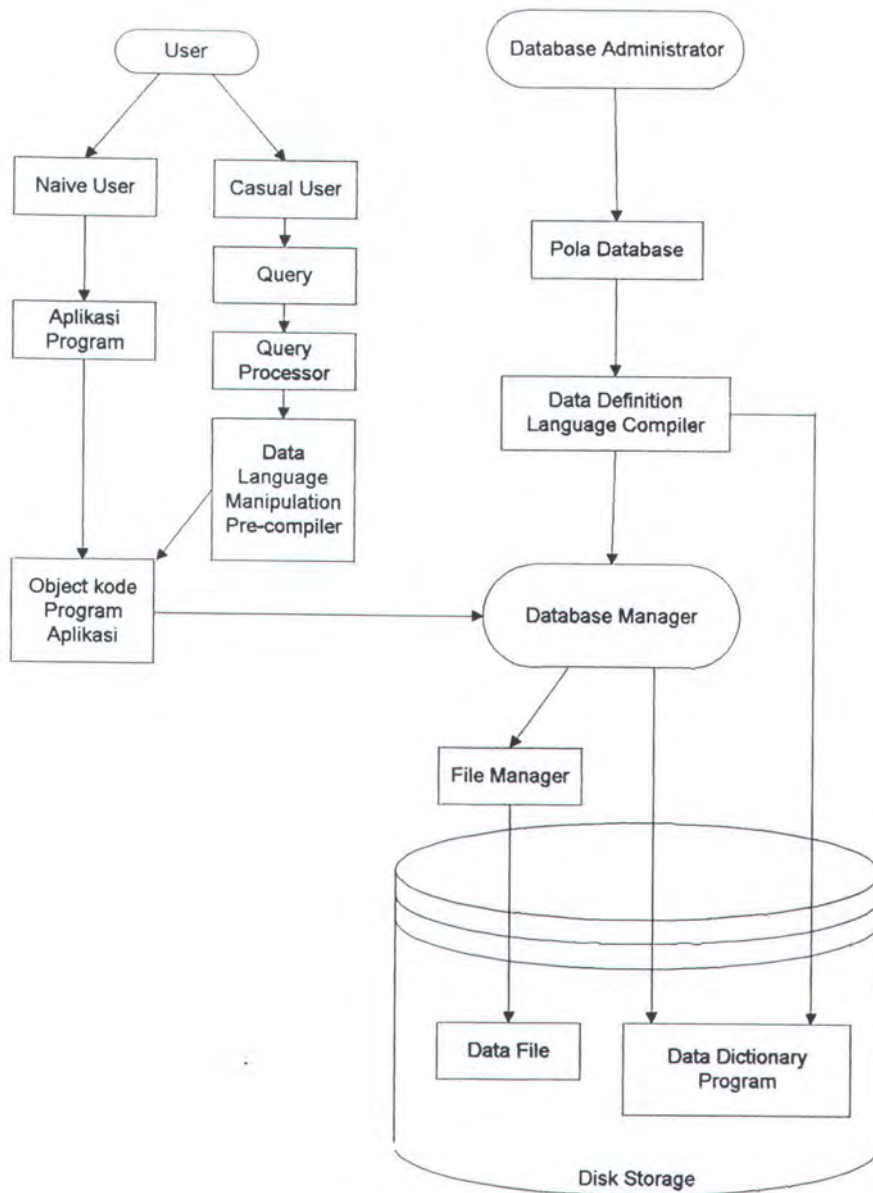
#### *Casual User*

Pemakai program *database* maupun bahasa pengolah *database* yang telah berpengalaman. *Cassual User* ini berinteraksi dengan program melalui bahasa *query* yang belum ada ( belum dibuat oleh programmer ).

#### *Naïve User*

Pemakai program *database* maupun bahasa pengolah *database* yang belum berpengalaman. Ia berinteraksi dengan program maupun bahasa pengolah *database* melalui pemilihan menu yang telah ada ( telah dibuat oleh programmer ).

Hubungan antar pengguna *database* dan struktur *databasenya* dapat dilihat dalam gambar dibawah ini :



## PERANCANGAN DATABASE

Untuk merancang suatu *database* diperlukan suatu keahlian dan pengetahuan tentang *database* itu sendiri. Hal ini karena merancang *database* merupakan pekerjaan yang sangat sulit. Kesulitan utama dalam merancang

*database* adalah bagaimana merancang pola *databasenya* sehingga dapat memuaskan pemakai / *user* sesuai dengan yang diinginkan. Untuk itu perancangan model konseptual akan menunjukkan *entity* dan relasinya berdasarkan proses yang diinginkan oleh organisasi perusahaan. Ketika menentukan *entity* dan relasinya dibutuhkan analisis data tentang informasi yang ada dalam spesifikasi di masa mendatang. Semua ini adalah tugas seorang *Database Administrator*. Pada perancangan model konseptual penekanan dilakukan pada struktur data dan struktur program.

Adapun pendekatan yang dilakukan pada model konseptual ada dua cara yaitu :

- Teknik Normalisasi
- Teknik *Entity Relationship*

Kedua cara diatas akan dijelaskan sebagai berikut :

### **TEKNIK NORMALISASI**

Teknik ini dipakai dalam pembuatan program untuk mengelola satu *file database* tanpa ada hubungan dengan *file* yang lain dalam satu *database*. Jadi teknik ini hanya dipakai untuk satu *file database* saja. Dalam pelaksanaannya teknik Normalisasi ini dikenal dengan 2 jenis kunci yaitu :

#### **FIELD KUNCI ( KEY FIELD ) :**

*Field* kunci adalah suatu *field* dalam *file* yang dapat mewakili suatu *record* tertentu. Setiap pencarian *record* cukup dengan menyebut salah satu dari *field* diatas maka akan didapatkan data lengkap mengenai *record* yang diinginkan. Namun ada kemungkinan bahwa ada beberapa *record* yang diwakili oleh salah satu kunci *field*. Karena itu kunci *field* kurang efektif bila dipakai untuk mengidentifikasi suatu kejadian spesifik dari *entity* dalam mencari



suatu *record*.

### **FIELD KUNCI KANDIDAT ( KEY CANDIDATE FIELD ) :**

*Field* kunci kandidat adalah satu kunci *field* yang dapat mengidentifikasi secara unik suatu kejadian spesifik dari suatu *entity*. Karena kekhusussannya, *field* kunci kandidat dapat digunakan untuk mencari *record* tertentu yang mempunyai ciri-ciri berbeda dengan *record* yang lain, baik itu melalui perintah *SEARCH*, *LOCATE*, maupun *FIND*.

Karena itu teknik Normalisasi ini digunakan hanya untuk mengorganisasi *file* saja, maka teknik ini belumlah merupakan teknik pemrograman *database* yang dapat memberikan hasil optimal bagi pemakai / *user*.

### **TEKNIK ENTITY RELATIONSHIP :**

Teknik *Entity Relationship* digunakan untuk pembuatan program pengelola *file-file* yang saling berkaitan. Teknik ini memerlukan suatu *Field Kunci Relasi ( Relation Field Key )*, yaitu *field* kunci yang terdapat dalam *file-file* dalam suatu *database*. *Field* ini menjadi *field* penghubung antar *file*. Sehingga bila programmer menghapus *field* kunci ini maka *field* ini akan hilang dari *file-file* yang memuatnya. Ada beberapa jenis hubungan antar *file* dalam 1 *database* yang ditentukan oleh jenis *field* kunci relasinya yaitu :

#### **One to one relationship :**

*One to one relationship* adalah hubungan antara dua *file* yang berbeda dalam satu atau lebih *database*. Hubungan ini diwujudkan dengan penentuan satu atau lebih *Field Kunci Relasi* sehingga pengontrolan *record* pada 2 *file* tersebut dapat dilakukan melalui *field kunci relasi* tersebut. Misalnya

programmer ingin membuat *query* pada 2 *file* yang terhubung pada *field kunci relasi* tertentu, maka ia akan mendapatkan hasil proses *query* sesuai kriteria dari kedua *file* yang bersangkutan.

***One to many relationship :***

*One to many relationship* adalah hubungan antara lebih 2 *file* yang berbeda dalam satu atau lebih *database*. Hubungan ini diwujudkan dengan penentuan satu atau lebih *field kunci relasi*. Bila dilakukan sesuatu proses akan memberikan hasil yang berhubungan dengan *file-file* yang terkait tadi. Jadi apabila diadakan perubahan pada *field kunci relasi* tersebut, maka data *value* dari *record* yang didalamnya terdapat *field* tersebut akan berubah. Untuk itu penerapan teknik *Entity Relationship* ini diperlukan pengalaman dan harus hati-hati dalam memakainya, karena suatu kesalahan yang tak disengaja yang dilakukan pada *field kunci relasi* akan mengakibatkan perubahan data *record file-file* yang terkait. Namun bagaimanapun teknik *Entity Relationship* ini sangat berguna dan akan memberikan hasil yang optimal bila untuk merancang program *database* yang mempunyai banyak *file* di dalamnya.

**SURVEI MENGENAI BENTUK PENYIMPANGAN  
PADA PROSES PRODUKSI KAPAL**

Oleh :

**TEDDY KRISHNA FEBION**

TEKNIK PERKAPALAN - FAKULTAS TEKNOLOGI KELAUTAN  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER SURABAYA

1997



Pilihlah jawaban yang anda anggap paling benar.

1. Pada proses produksi kapal sering terjadi penyimpangan terhadap bentuk dan ukuran dari perencanaan, diantara penyimpangan dibawah ini penyimpangan manakah yang sering terjadi :
  - a) Deformasi sudut
  - b) Deformasi memanjang
  - c) Deformasi buckling
  - d) Penyusutan
  - e) Kesalahan pemasangan
2. Pada tahap proses produksi apakah penyimpangan sering terjadi:
  - a) tahap Fabrikasi
  - b) tahap Sub-Assemble
  - c) tahap Assemble
  - d) tahap Erection
  - e) tahap lain
3. Pada daerah manakah yang paling sering terjadi ketidak lurusan ( misalignment ) pada proses produksi kapal :
  - a) Daerah lambung
  - b) Daerah geladak
  - c) Sekat melintang
  - d) Sekat memanjang
  - e) Pelat alas dalam
4. Diantara konstruksi dibawah ini, daerah manakah yang sering mengalami perubahan sudut :
  - a) Antara konstruksi lambung dan Geladak.
  - b) Antara pelat alas dalam dan sekat melintang.
  - c) Antara pelat alas dalam dan pelat lunas.
  - d) Antara pelat alas dalam dan sekat memanjang.
  - e) konstruksi lain.

5. Diantara penyimpangan dibawah ini, penyimpangan manakah yang paling sering terjadi pada konstruksi lambung :
- a) Perubahan sudut antara konstruksi lambung dan Geladak.
  - b) Terjadi penggelombangan pada pelat sekat
  - c) Terjadinya defleksi pada block lambung.
  - d) Terjadinya ketidak tepatan pada sambungan pengelasan ( seams lag ) pada pelat alas dalam pelat sekat.
  - e) Terjadi penyimpangan bentuk lain.
6. Faktor apakah yang sering menjadi penyebab utama terjadinya penyimpangan bentuk pada proses produksi kapal.
- a) Faktor Material.
  - b) Faktor Tenaga Kerja.
  - c) Faktor Metode produksi.
  - d) Faktor Peralatan produksi.
  - e) Faktor lain.
7. Pada proses erection penyimpangan apakah yang sering dijumpai :
- a) Terjadi ketidak lurusan pada pertemuan sambungan antara block-block.
  - b) Terjadinya gab-gap pada sambungan tumpul diantara block-block.
  - c) Terjadinya distorsi selektif pada block lambung.
  - d) Terjadinya defleksi pada block lambung.
  - e) Terjadi penyimpangan lain.
8. Diantara penyimpangan dibawah ini, penyimpangan manakah yang sering terjadi :
- a) Terjadinya lekuk-lekuk pada centre girder.
  - b) Terjadinya lekuk-lekuk pada side girder.
  - c) Terjadinya lekuk-lekuk pada penumpu-penumpu.
  - d) Terjadinya penggelombangan pada pelat sekat.
  - e) Terjadi penyimpangan lain.

9. Pada suatu proses produksi kapal sering terjadi penyimpangan yang disebabkan oleh kasalahan atau kurang telitian pengukuran, daerah manakah yang paling sering mengalami penyimpangan yang disebabkan oleh kasalahan tersebut :

- a) Daerah pelat kulit.
- b) Daerah pelat sekat memanjang.
- c) Daerah pelat lunas.
- d) Daerah pelat alas dalam.
- e) Daerah sekat melintang.

10. Pada setiap proses produksi, pada daerah manakah yang sering mengalami deformasi

- a) Sambungan tumpul, yaitu pada pelat kulit , pelat alas dalam dan pelat sekat.
- b) Pada Centre girder, Side girder dan Penumpu-penumpu.
- c) Pada pelat kulit dan pelat geladak.
- d) Pada pelat alas dalam dan sekat melintang.
- e) Pada daerah lain.



Berilah tanda  $\checkmark$  pada kolom jawaban yang anda anggap paling benar

	YA	TIDAK
1. Pada proses erection sering terjadi ketidak lurusan pada pertemuan sambungan antara block-block atau seksi-seksi.	<input type="checkbox"/>	<input type="checkbox"/>
2. Pada proses erection sering terjadi gap-gap pada sambungan tumpul diantara block-block atau seksi-seksi.	<input type="checkbox"/>	<input type="checkbox"/>
3. Pada sambungan T antara hubungan sekat melintang dengan pelat geladak sering terjadi gap.	<input type="checkbox"/>	<input type="checkbox"/>
4. Apakah sering terjadi perubahan sudut antara konstruksi lambung dan geladak.	<input type="checkbox"/>	<input type="checkbox"/>
5. Apakah sering terjadi penyimpangan sudut antara pelat alas dalam dan pelat lunas.	<input type="checkbox"/>	<input type="checkbox"/>
6. Apakah sering terjadi penyimpangan sudut antara pelat alas dalam dengan sekat melintang.	<input type="checkbox"/>	<input type="checkbox"/>
7. Apakah deformasi sudut sering menyebabkan terjadinya penyimpangan pelat antara dua titik tumpuannya.	<input type="checkbox"/>	<input type="checkbox"/>
8. Apakah deformasi buckling sering menyebabkan terjadinya lekuk-lekuk pada centre girder, side girder dan penumpu- penumpu.	<input type="checkbox"/>	<input type="checkbox"/>

9. Apakah deformasi sudut sering menjadi penyebab terjadinya angular misalignment pada sambungan tumpul, yaitu pada pelat kulit, pelat alas dalam, dan lain-lain.

☐ ☐

10. Deformasi memanjang sering menjadi penyebab terjadinya defleksi pada block lambung.

☐ ☐

11. Terjadinya ketidak lurusan pada pertemuan sambungan antara block-block atau seksi-seksi sering disebabkan adanya penyusutan pada block atau seksi pada saat proses pembuatannya.

☐ ☐

12. Apakah deformasi buckling dan perubahan sudut pada penegar-penegar sering menjadi penyebab terjadinya penggelombang pada pelat kulit.

☐ ☐

13. Adanya penyusutan, deformasi sudut, deformasi memanjang dan kesalahan pemasangan sering menjadi penyebab terjadinya penyimpangan kearah kelebaran dari badan kapal.

☐ ☐

14. Apakah adanya deformasi memanjang, deformasi sudut, dan kesalahan pemasangan menjadi penyebab timbulnya penyimpangan sudut antara pelat alas dalam dan sekat melintang.

☐ ☐

15. Terjadinya perubahan sudut antara konstruksi lambung dan geladak banyak disebabkan karena adanya perubahan sudut, penyusutan dan deformasi memanjang.

☐ ☐

16. Apakah deformasi sudut dan kesalahan pemasangan sering menjadi penyebab terjadinya perubahan sudut antara pelat alas dalam dan pelat lunas.

☐ ☐

17. Deformasi sudut sering menjadi penyebab terjadinya distorsi relatif pada block lambung.

☐ ☐

18. Pada proses erection sering terjadi ketidak tepatan pada sambungan pengelasan ( seamlag ) pada pelat alas dalam, pelat sekat dan lain-lain.

☐ ☐

19. Pada tahap erection sering dijumpai adanya lekuk pada kulit dan sekat memanjang / melintang.

☐ ☐

20. Apakah sering terjadi gap pada sambungan tumpul pada bracket.

☐ ☐

21. Deformasi sering menjadi penyebab terjadinya penyimpangan pada proses produksi kapal.

☐ ☐

22. Apakah sering terjadi lekuk pada pelat kulit, sekat dan pelat lunas.

☐ ☐

23. Pernahkah terjadi ketidak lurusan pada sambungan senta antara dua block yang akan disambung.

☐ ☐

24. Apakah bentuk penyimpangan seperti pada soal no 25 sering terjadi pada setiap proses produksi kapal.

☐ ☐



25. Masih adakah bentuk penyimpangan lain selain yang ada dibawah ini :

- Deformasi sudut
- Deformasi memanjang
- Deformasi buckling
- Penyusutan
- Kesalahan pemasangan



## **PENYIMPANGAN PADA PROSES FABRIKASI**

<b>DIMENSI</b>		<b>PENANDAAN (Bevel / Okobei)</b>
<b>General Members dibanding dengan Ukuran yang Benar</b>	<b>Terutama Tinggi Wrang dan Penumpu Dasar Ganda dibanding dengan Ukuran yang Benar</b>	
<ul style="list-style-type: none"> <li>• Pada proses nesting terjadi kesalahan pada potongan 2 komponen dimana hasil potongan memotong kedua komponen sepanjang 40 mm. ( gambar dan program sudah benar ).</li> <li>• Terjadi kesalahan dimensi (tekor / terjadi gap) pada material pelat sebesar 15,40 dan 50 mm.</li> <li>• Terjadi ketidak cocokan dimensi lebar antara program dan nesting dari hasil pemotongan NC plasma.</li> </ul>	<ul style="list-style-type: none"> <li>• Terjadi kesalahan program yang mengakibatkan material mengalami tekor / gap sebesar 8 mm.</li> </ul>	<ul style="list-style-type: none"> <li>• Terjadi kesalahan penandaan bevel, tanda bevel terbalik.</li> <li>• Terjadi kesalahan penandaan pada posisi IN dari design. Tanda bevel terbalik.</li> </ul>

## PENYIMPANGAN PADA PROSES ASSEMBLY

DIMENSI	KETELITIAN PEMASANGAN						DISAIN
	Gap Sebelum Pengelasan				Gap Antara Bagian Komponen		Kesalahan Pemasangan
Plat Block	Plate Deck	Bottom Shell	Tank Top & Plate Shell	Plate Shell	Wing Bulkhead	Frame	
<ul style="list-style-type: none"> <li>Block RD plate 24 hasil tidak sesuai dengan gambar kerja. Ukuran yang benar 254 mm, sedangkan lebar actual 390 mm, jadi tidak cocok dengan plate dalamnya.</li> </ul>	<ul style="list-style-type: none"> <li>Pada pelat Deck posisi Aft lebar 2750, keadaan actual 2650, jadi terjadi gap sebesar 100 mm.</li> </ul>	<ul style="list-style-type: none"> <li>Pada block Double Bottom terjadi tekor / gap sebesar 20 mm terhadap Bottom Shell.</li> <li>Terjadi tekor/gap sebesar 50 mm terhadap Bottom Shell (Bilga).</li> </ul>	<ul style="list-style-type: none"> <li>Terjadi tekor / gap antara Tank Top ke Joint Block / Shell Joint sebesar 15 - 30 mm.</li> </ul>	<ul style="list-style-type: none"> <li>Pada Block Double Bottom sambungan antara dua plat shell terjadi gap sebesar 12 mm.</li> <li>Pada Join Plat Shell terjadi gap sebesar 15 - 35 mm.</li> </ul>	<ul style="list-style-type: none"> <li>Terjadi gap antara sambungan Wing Bhd antara 13 - 43 mm.</li> <li>Pada pelat Wing Bhd marking penempatan Stifener dari nesting tidak cocok dengan lubang slot, serta lebar Wing Bhd terjadi gap sebesar 40 mm.</li> <li>Pada komponen Wing Bhd potongan NC Cutting antara tiga jarak frame</li> </ul>	<ul style="list-style-type: none"> <li>Terjadi kesalahan dimensi pada frame dimana komponen mengalami tekor antara 12 - 20 mm.</li> <li>Pada salah satu frame (P/S), terjadi gap sebesar 30 mm terhadap Tank Top.</li> </ul>	<ul style="list-style-type: none"> <li>Lubang longitudinal untuk posisi Wing Bhd dalam kenyataan adalah sebesar 160 mm, sedang seharusnya adalah 200 mm.</li> <li>Pengelasan pada Frame tidak bisa dilakukan karena Man-Hole tidak ada pada Tank Top.</li> <li>Pada Main Deck ada lubang sebesar 750 x 675 p/s, sedangkan pada gambar tidak ada lubang.</li> <li>Pada salah satu frame, materialnya ada lubang Scalof semuanya (P/S).</li> </ul>



					<p>berbelok keatas sehingga terjadi gap sebesar 10 mm.</p> <ul style="list-style-type: none"> <li>• Lebar komponen gambar 1000 mm, sedangkan material komponen lebar 950 mm. Jadi terjadi tekori sebesar 50 mm.</li> </ul>		<p>seharusnya pada bagian (P) tertutup.</p> <ul style="list-style-type: none"> <li>• Pada salah satu frame materialnya tidak ada lubang Scalof semua bagian (P/S).</li> <li>• Pada salah satu frame ada lubang longitudinal, sedangkan gambar tidak ada lubang longitudinal.</li> <li>• Marking penempatan side longitudinal tidak ada sehingga dilapangan kesulitan Fitting longitudinal pada Wing Bhd.</li> </ul>
--	--	--	--	--	--	--	---

## PENYIMPANGAN PADA PROSES ERECTION

DIMENSI		KETELITIAN PEMASANGAN		DISAIN		
Kerataan Pelat	Deformasi	Gap Sebelum Pengelasan		Gap Antara Bagian Komponen	Pemotongan Okobei	Kesalahan Pemasangan
		Deck Plate	Shell Plate			
<ul style="list-style-type: none"><li>• Terjadi kesalahan dimensi pada Bracket.</li><li>• Terjadi kekurangan dimensi pada Stiffener sebesar 35 mm (P) dan 55 mm (S).</li><li>• Terjadi ketidaksesuaian antara Wing Bulkhead pada Transverse, besar penyimpangan sebesar 30 - 45 mm. Kesalahan pada lebar joint block bagian bawah (drawing sebesar 1600 mm), yaitu kurang dari 1600.</li></ul>	<ul style="list-style-type: none"><li>• Terjadi Deformasi pada plat tepi yang terletak pada sisi kanan (S).</li></ul>	<ul style="list-style-type: none"><li>• Tinggi sambungan pelat dari wash Deck kurang sebesar 150 mm dari yang direncanakan pada Drawing.</li><li>• Pada sekat melintang sambungan antar plat tersebut mengalami Misalignment sepanjang ± 350 mm dan terjadi gap pada deck sepanjang + 300 mm dan tinggi 20 mm.</li></ul>	<ul style="list-style-type: none"><li>• Terjadi Gap antara sambungan dua Plat Shell Block antara 8 - 20 mm.</li><li>• Plat Shell Block DB terjadi gap sebesar antara 0 - 15 mm dan Plat Shell Block lain antara 0- 10 mm.</li><li>• Plat Shell Block DB terjadi gap sebesar antara 0 - 17 mm dan Plat Shell Block lain antara 0- 9 mm.</li><li>• Pada Block terjadi gap sebesar 3, 5 mm ke arah kanan (S).</li></ul>	<ul style="list-style-type: none"><li>• Hull Construction pada Engine Girder terjadi gap pada sambungan tumpul antar floor dan engine girder.</li><li>• Terjadi gap sambungan tumpul pada sambungan Bracket melebihi 10 mm.</li><li>• Terjadi misalignment untuk prose joint erection pada upper deck.</li><li>• Terjadi gap pada sambungan T antara floor dan side girder double bottom.</li></ul>	<ul style="list-style-type: none"><li>• Terjadi kesalahan pemotongan tirus (Okobei) pada sambungan dua pelat.</li><li>• Terjadi kesalahan sambungan dua pelat yang mempunyai ketebalan 19 mm dan pelat dengan ketebalan 21 mm seharusnya sambungan keduanya tanpa okobei</li><li>• Terjadi kesalahan pada sambungan dua pelat dimana pelat pertama dengan tebal 21 mm dan pelat kedua dengan tebal 12 mm</li></ul>	<ul style="list-style-type: none"><li>• Lubang Sea Cest 625 x 600 pada pelat belum dipotong.</li><li>• Terdapat dua buah Main Hole 600 x 500 pada Wing Bhd diantara dua frame, seharusnya Main Hole tersebut tidak ada karena Wing Bhd kedap.</li><li>• Terdapat ludang Scalop, seharusnya kedap.</li><li>• Terdapat Tightening Hole dan Scalop, seharusnya kedap</li></ul>

<ul style="list-style-type: none"> <li>• Terjadi lekuk pada plat kulit yang dalam</li> </ul>					<p>belum dipotong tirus (Okobei).</p> <ul style="list-style-type: none"> <li>• Terjadi kesalahan pemotongan Okobei pada sambungan pelat, seharusnya kampuh tanpa Okobei karena mempunyai ketebalan pelat yang sama. Hal ini sering terjadi pada penebalan pelat di daerah Sea Cest.</li> <li>• Terjadi ketidaksesuaian pada sambungan pelat, hal tersebut disebabkan oleh panjang pelat yang seharusnya bersesuaian tidak sama. Karena pelat-pelat tersebut mempunyai perbedaan ketebalan yang cukup besar sehingga bevel dan Okobei menjadi tidak sesuai.</li> </ul>	<p>karena membatasi tangki.</p> <ul style="list-style-type: none"> <li>• Karena kondisi di lapangan, maka dibuat Temporary Man Hole pada pelat Inner Bottom untuk kelancaran pengerjaan block.</li> <li>• Tangki terletak diantara 4 jarak Frame, Joint Block belum dapat di Fitting disebabkan lubang Man Hole yang terletak disalah satu Frame tidak ada, sehingga ruangan tidak dapat dimasuki.</li> <li>• HP 370 x 13 pada salah satu</li> <li>• frame menutupi Man Hole Tank Top, sehingga Man Hole tidak berfungsi.</li> </ul>
--	--	--	--	--	---	--